

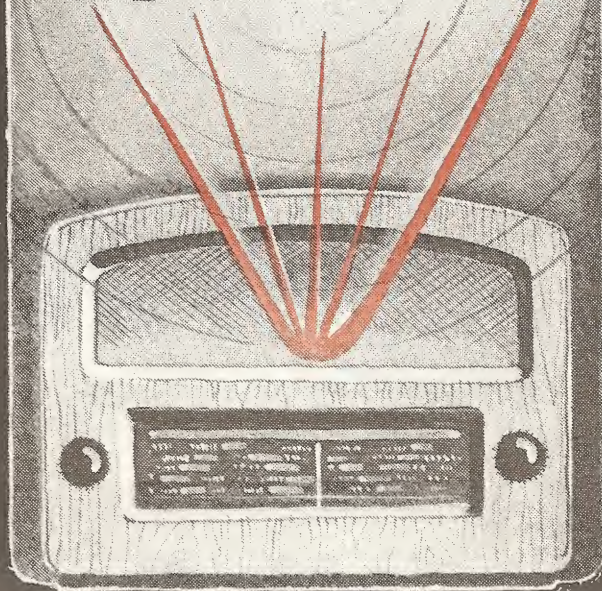


Mármint számítógépes programokat? A kérdőjel mondhatnák nem jogos, hiszen a dolog már kipróbálásra került, több mint sikerrel.

Kétségtelen, hogy ha emlékeznek, alig három hónapja, éppen az október végén megjelent BIT-LET-ben vertük a mellünket, közölve, hogy talán épp a BIT-LET Karácsonyon kerül majd kipróbálásra a számítógépes programok rádióadás útján való terjesztése. Igaz, a mellverésről mire megjelent kiderült, hogy idejétmúlt, hiszen a lap megjelenése után néhány nappal már a tv és a rádió egyaránt közvetített egy kétórás kísérleti adást, de meg kell mondanom, hogy mindebben legfeljebb az bántott bennünket, hogy téves hírek mondói jégre vittek bennünket, s ily módon kicsit nevetségessé váltunk mell-döngtésünkkel. Az így azonban fontosabb volt, mintsem ez érdekelhetett volna különösképpen bennünket. Nosza rá is vettünk néhány ismerőst, hogy a főváros különböző pontjain vegyék föl az adást és próbálják ki. Nos, ezúton is jelenthetjük, hogy a felvételek nagyobbik része sikerült. Így hát nagy örömeinkre szolgált, hogy a BIT-LET Karácsony rendezvényére is eljöttek a Posta szakemberei, s ott azután két napig nyomták az éterben a programhullámokat.

No, de haladjunk időrendi sorrendben. Néhány nappal a BIT-LET Karácsony előtt, a szerkesztőt meghívták a rádió Reggeli csúcs című műsorába, meséljen pár szót erről a nagyszabású örületről, ami ott a Műegyetemem készül. A mesélés végén pedig közös kísérletünk gyanánt leadtunk egy programot, éspedig egy C 16-os programot, hadd vegyék ezt azután már országszerte. Kértük, hogy tudassák is velünk az olvasók, hallgatók, hogy a felvételek hogy sikerültek. Nem mondom, hogy sok visszajelzés érkezett, de azért jött. Azt sem állítanám, hogy az eredmény meggyőző volt, hiszen vidéki felvétel is volt, hogy működött, s budapesti is, amelyik nem. Úgy tűnik tehát, hogy a nem postai recept szerint sugározott,

Sugározunk?



csak úgy az éterbe lőkött programok nem igazán biztonságosan vehetők. Hogy a posta szakemberei mit csinálnak a jellel, ezt nem árulták el, mindenesetre mi azt az egy tapasztalatunkat megosztjuk olvasóinkkal, hogy a némileg túlvezérelt felvételnek nagyobb esélye volt a beolvasáskor, mint a helyesen modulált felvételnek. S képzeljék!

Olyan levél is jött, amelynek írója a tévé hangszórójáról, mikrofonnal készített felvételt, s az működött!!!

Szinte hihetetlen! Egy szó, mint száz, a kísérletek végére valahára elkezdődtek. Igaz, régi jó szokás szerint elsősorban a fővárosi amatőröknek van lehetőségük részt venni benne, s a vidékiek csak az ilyen reggeli csúcsban elszórt kalózádasokkal kacérkodhatnak. Öszintén szólva bennünket nagyon érdekelne, hogy a kísérletből mikor lesz már adás!

S hogy kinek, milyen elképzelése van arról, hogy mi megy majd az éterből a számítógépbe? Mert a legbonyolultabb Spectrum,

vagy C 16 program is belefér 5-10 perche! Márpedig a nap 24 órából áll. Hogy mire lehet egyáltalán ezt a kommunikációs eszközt használni? Hát kérem, lehet vele marhaságokat is közvetíteni. Eddig jobbra ebbe a kategóriába tartozó dolgok keltek szárnyra ily módon, hiszen a kísérletben részt vevőknek arra már igazán nem volt energiájuk (no meg pénzük), hogy valami értelmes programot írassanak a vevőkészülékek előtt ülőknek. De azért ha belegondolnak mondjuk a távoktatás és a számítógép, valamint a rádióhullámok terjedési sebessége hármass összefüggésébe, nos alighanem csettintenek egyet a végeredményt elképzelve. Ja, hogy mikor lesz az? Nos, ezt nem tudjuk. Egyet azonban biztosan. A sugározgatás kicsit későn, mondjuk úgy: még Kelet-európai mértékkel mérve is későn kezdődött el nálunk. Ha ugyanebben a tempóban folyik tovább a fejlődés, akkor talán még unokáink sem vitatkozhatnak azon, hogy mire jó a computer a rádióhullámokon.

Angyalosi László

BELÜLRŐL

- 18 **A szokásos Hiroidal** – a szokásos érdekességekkel, s egy új gép fotójával
- 20 **Életjáték** – a C 16-os program többeket arra indított, hogy saját gépükre is írjanak hasonló programot. Két működőképes sejtautomata Spectrumra és Primora.
- 22 **A Beta Basic hibái** – egy Spectromos BASIC bővítésről, a hibák miértjéről
- 23 **Programajánlat** – egy C 16-os program, amely olyasmire tanítja a gépet, amit csak a Sinclair tudott
- 24 **Siker volt!** – képriport a BIT-LET Karácsonyról
- 26 **Első kézből a tv-computerről** – megkezdett sorozatunk folytatásaként, ezúttal a beépített videó rutinokról, megfelelő példaprogrammal
- 28 **Könyvmoly** – amelyben ezúttal nemcsak az elmúlt 1-2 hónapban megjelent könyvek listáját közöljük, de igyekeztünk összeállítani az 1986-os évben megjelent számítástechnikai könyvek minél teljesebb listáját is.
- 30 **Posta** – amelyben böstörög a Novotrade, válaszol a szakértő is...
- 31 **Kiegészítés a narancs díjhoz** – azaz a C 16 RE-NEW-hoz
- 31 **A gordiuszi csomó** – amelynek elvágása után a Junoszty tévén is megszólal a Commodore hangja
- 32 **Nyerő...nyerő...nyerő...** – eredményekkel, megoldással, és új feladattal

DISZPÉCSER

SUGÁRKAPCSOLAT

KAPCSOLAT

MŰZEUM
egyszámú
évre

OPTICOMPUTER

MŰVÉSZET

KÖZLEKEDÉS

18

ÉTREND

A mexikói vb-n elszenvetett vereségünk okait azóta is kutatjuk. A közelmúltban számítógéphez táplálták a magyar vb csapat kinti étrendjét, hogy az adott élelmezési költségek és étkezési szokások figyelembe vételével számítsa ki az étrend diétetikai jellemzőit. Megállapították, hogy Mexikóban az ellátás számszerűsített, a 70-80 kg-os test-súlyúak részére megfelelő kalóriamennyiséget szolgáltatott. Kiderült, hogy a hazai népi élelmezési adatokat figyelembe véve a fentebb említett étrend alapján összeállított étrenddel. Végül is a számítógép az adatok szerint nem felel meg a valószínűsíthető kondíciómérési és a túlzott szénhidrát-fogyasztás okozta.

TERJESZKEDES

Meglepő elhatározásra szánta el magát a japán Fujitsu cég. Bejelentette, hogy megvásárolja a világhírű integráltáramkör gyártó vállalatot a kaliforniai Fairchild-et. A washingtoni kormány még valamilyen ürüggyel megvétőzhatja az üzletet. Ha az adás-vétel nyelvébe üttök, a japán cég jelentősen növelheti amerikai eladásait, hiszen az amerikai gyártókkal előállított termékeire a kormány nem kezdeményezhet dömpingvizsgálatot.

MÉRLEG

A hőmérsékletmérő Metripod Mérleggyárban megkezdte a vállalat saját konstrukciói által kifejlesztett, formatervezett kivitelű, mikroprocesszoros vezérlésű, árszorító bolti mérlegek sorozatgyártását. A három-, öt- és tízkilos mérlegek nemcsak az áru súlyát, hanem az érte fizetendő összeget is mutatják, s memóriagységük lehetővé teszi, hogy a napi forgalom összegzésével elvégezzék a pénztárgép feladatát is.

PONTOS VIZSGÁLAT

Az egyik londoni kórház orvosai és számítástechnikai szakemberei együttesen olyan számítógépes mellkasvizsgálati módszert dolgoztak ki, amely mentesíti a beteget a hagyományos vizsgálatok megterhelésétől. A dolog lényege, hogy a páciens felsőtestére fénypontokat vetítenek, majd a légzési ciklus alatt ezeket a pontokat két irányból lefényképezik. Az így kapott képeket, illetve az azokból származó adatokat számítógéppel kiértékelik, melyből megállapítható a belégzett levegő mennyisége, a tüdő átáramló vérmennyisége, a mellkas, a tüdő különböző megbetegedései.

FOCIERŐSZAK

Bár Európában az angol labdarúgó szurkolók a legveszélyesebbek, más országokban is egyre nagyobb gondot okoz a labdarúgó-mérkőzések erőszakmentes lebonyolítása, a rendezők erőszakmentes, kiszűrője. Ilyen problémákkal kell szembenézniük Hollandiában is a sporteseményeket rendezőknek és a sportesemények közönségének is. Ezért állított fel számítógépes adattárat egy utrechti intézet. A gépbe folyamatosan betáplálják a pályák rendbontóinak, erőszakoskodóinak adatait és ezzel segítik a rendezők és a rendőrség munkáját. Az adattárba néhány hónap alatt több mint ezer személy neve, adata került be.

IC TEMPO

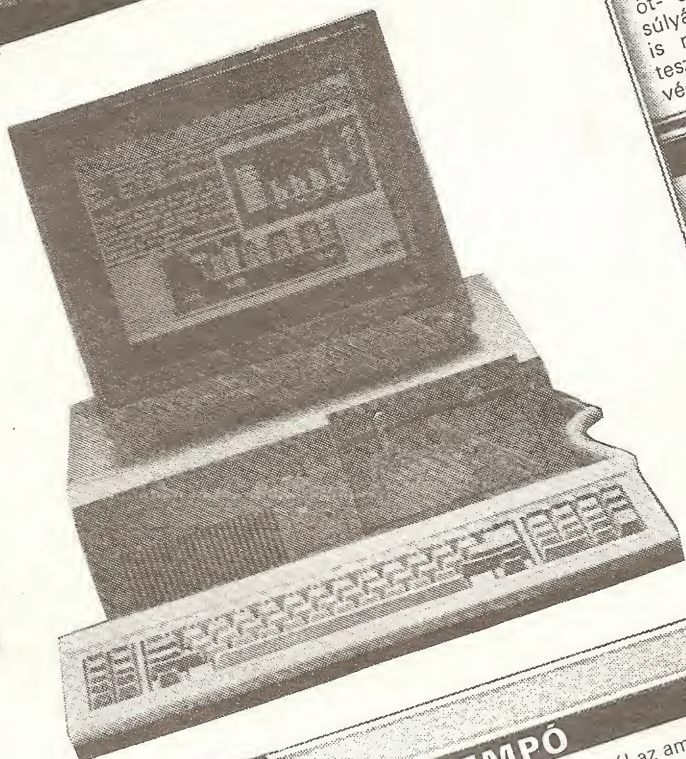
Szenzációs tempót diktál az amerikai IC technológiai ipar. A General Electric cég a konkurenciát áramkörök részére vállalta az igényelt, 24 órán belül 800-3000 kapuáramkört tartalmazó berendezésorientált integrált áramkörök (BOAK) sorozatgyártásának megkezdését. E nem mindennapi teljesítményt a General Electric azáltal érte el, hogy megvásárolta a Laseparth cég új lézeres maszkolási technológiáját.

BORSZAKÉRTŐ

Speciális, számítógépes borszakértő rendszert fejlesztettek ki a hollandiai Élelmiszer-ellenőrző Intézetben. Azon túlmenően, hogy a rendszer pillanatok alatt megállapítja a vizsgált borsfajta szesz- és cukortartalmát, a kémiailag összetétel alapján könnyen utólrétegzhető és termőterületére is. A holland számítógépes rendszerrel könnyen kimutatható a borhamisítók is. Ugyanis, amint anyagok a különböző „feljavító” vegyi anyagok jelenlétét, azt is megállapítja, hogy a hamisított bor honnan, mely termőterületről származik.

SPERRY PC/microIT

A Sperry cég bemutatta PC/IT számítógépénél kisebb, olcsóbb és gyorsabb új típusát, a PC/microIT gépet. Amíg a PC/IT maximális sebessége 7,16 MHz, az új számítógép 8 MHz-es mikroprocesszorral fut. A rendszer támogatja a szinkron és aszinkron kommunikációt. Konfigurációtól függően a PC/microIT-öt felhasználhat XENIX System V alatt. Lemezegység nélküli kiépítésben a gép 512 Kbyte-os memóriával és Key Tronic típusú klaviatúrával rendelkezik. Más kiépítésben 20 Mbyte-os hard diszk meghajtót tartalmaz, de floppyt nem.




```

10 BORDER 0: PAPER 0: INK 7: CLS
20 POKE USR "a",255: FOR i=1 TO 7: POK
E USR "a"+i,0: NEXT i
30 INK 0: LOAD "CODE": INK 7
40 PRINT AT 4,0:"Mennyi ido legyen Ket
generacio kozott?(masodpercben)"
50 INPUT "ido:";ido
60 CLS
70 PRINT AT 2,10:"FUNKCIOK:" PRINT AT
3,10: OVER 1:
80 PRINT AT 5,8:"5.....balra"
90 PRINT AT 6,8:"6.....jobb"
100 PRINT AT 7,8:"7.....le"
110 PRINT AT 8,8:"8.....fel"
120 PRINT AT 9,8:"9.....elet-halal"
130 PRINT AT 12,0:"Futtatas Kozben '0'-
val lehet a menuhoz visszaterni."
140 PRINT AT 15,3:"Kerem varjon egy Kic
siti!"
150 PRINT AT 10,8:"ENTER....vege"
160 PRINT AT 17,0:"Szerkesztes Kozbeni
'1'-gyel, 'il'-letve '2'-vel lehet a genera
ciot elmenti es visszahivni."
170 PRINT "A vegrehajtast '+' jelzi a b
al felso sarokban."
180 RANDOMIZE USR 43150
190 PRINT AT 15,1: FLASH 1:"Nyomjon le
egy billentyut!"
200 PAUSE 0: LET a$=INKEY$: IF a$="" TH
EN GO TO 200
210 CLS
220 PRINT AT 10,10: FLASH 1: " "
230 LET x=10: LET y=10
240 PAUSE 0: LET a$=INKEY$: IF a$="" TH
EN GO TO 240
250 PRINT AT 0,0: " "
260 IF a$=CHR$ 8 OR a$="5" THEN GO SUB
0650: GO TO 350: REM bal
270 IF a$=CHR$ 9 OR a$="8" THEN GO SUB
0650: GO TO 390: REM jobb
280 IF a$=CHR$ 10 OR a$="6" THEN GO SU
B 0650: GO TO 420: REM le
290 IF a$=CHR$ 11 OR a$="7" THEN GO SU
B 0650: GO TO 450: REM fel
300 IF a$="0" THEN GO TO 480
310 IF a$=CHR$ 13 THEN GO TO 520
320 IF a$="1" THEN RANDOMIZE USR 43195
: PRINT AT 0,0: "+"
330 IF a$="2" THEN PRINT AT 0,0: "": RA
NDOMIZE USR 43210: PRINT AT 0,0: "+": PRI
NT AT y,x: OVER 1: FLASH 1: " "
340 GO TO 240
350 REM balra
360 LET x=x-1: IF x=0 THEN LET x=30
370 PRINT AT y,x: FLASH 1:CHR$ PEEK (40
960+32*y+x)
380 GO TO 240
390 REM jobbra
400 LET x=x+1: IF x=31 THEN LET x=1
410 GO TO 370
420 REM le
430 LET y=y+1: IF y=21 THEN LET y=1
440 GO TO 370
450 REM fel
460 LET y=y-1: IF y=0 THEN LET y=20
470 GO TO 370
480 REM rakas
490 IF PEEK (40960+32*y+x)=42 THEN POK
E (40960+32*y+x),32: GO TO 510
500 POKE (40960+32*y+x),42
510 GO TO 370
520 PRINT AT y,x:CHR$ PEEK (40960+32*y+
x)
530 LET gen=1
540 PRINT AT 0,0:gen:".generacio "
550 PAUSE 50*ido
560 RANDOMIZE USR 43000
570 LET gen=gen+1
580 PRINT AT 0,0:gen:".generacio "
590 LET tag=USR 43170
610 PRINT AT 0,15:tag:" sejt "
615 BEEP .05,30
617 PAUSE 100*50
620 IF tag=0 THEN GO TO 0660
630 LET a$=INKEY$: IF a$="0" THEN GO T
O 60
640 GO TO 560
650 LET s=PEEK (40960+32*y+x): PRINT AT
y,x:CHR$ s: RETURN
660 CLS
670 PRINT AT 10,0:"Kihalt a ".gen:". ge
neracioban"
680 PRINT AT 14,0:"A menuhoz '0'-val le
het vissza- terni."
690 PAUSE 0: LET t$=INKEY$: IF t$="0" T
HEN GO TO 60
700 GO TO 690
8999 STOP
9000 SAVE "eletjatek" LINE 10
9010 SAVE "mcode"CODE 43000,230
9020 PRINT "ready"

```

ÉLETJÁTÉK



Októberi számunk felét tette ki az Életjátékról, sejtautomatákról szóló cikk és C 16-os program. Komoly dicséretnek érétkeltük egyik olvasónk levelét, aki közölte, bepötyögte a tekintélyes méretű programot, s legnagyobb megdöbbenésére első kísérletre gond nélkül futott! Ez igen! Tette hozzá. Ez igen nagy és körülményes ráadásul hosszan tartó – előkészítő munka eredménye volt – tehetjük mi hozzá. (Csak zárójelben jegyezzük meg, hogy az anyaghoz még tartozott egy tekintélyes méretű igen jól megszerkesztett folyamatábra is, amelyet vérző szívvel, de kihagytunk – terjedelmi okokból.) Nos, felhívásunk, hogy a témát szívesen folytatjuk – nem volt pusztába kiáltott szó. Erkezett két használható program, amelynek eredményeképpen most már a Spectrumosok és a Primosok is megismerkedhetnek az életjátékkal. Ha nem olvasták, úgy figyelmükbe ajánljuk az októberi cikk elméleti, történeti bevezetőjét!

SPECTRUMRA

A program egy BASIC részből és egy 224 byte-os gépi kódú csomagból áll. A BASIC rész szervezi a működést, de gépi kódban van megírva az érdemi rész. (Azaz azok a részek, amelyek BASIC-ben igen lassan futnának.)

A BASIC rész:

A program tartalmazza a futtatáshoz szükséges összes információt. A BASIC rész végzi el a kiindulási állapot bevitelét a kurzormozgató billentyűk és a "0" segítségével. Az "1"-es gombbal el lehet menteni az éppen képernyőn levő állapotot, és "2"-es gombbal vissza lehet hívni. "ENTER" lenyomásával lehet befejezni a kezdő állapot létrehozását és elindítani az adott populációt a fejlődés útján. Innen vagy a populáció kihalásával vagy pedig a "0" gomb lenyomásával lehet visszatérni a menühöz.

A gépi kódú rész:

5 programrészből áll, mindegyik egy speciális funkciót valósít meg. 43000-43147: ez a fő rutin. Ez generálja az új generációt, és rajzolja ki a képernyőre. Úgy működik, hogy az aktuális generáció minden egyes cellájáról eldönti, hogy életben lesz-e, vagy nem. Ezt a 41664-es címtől kezdve letárolja, majd minden cella végignézése után visszairja az alappufferbe, a 40960-as címtől kezdődően, majd kirajzolja. 43150-43166: inicializáló rutin. Space-szel tölti fel az alap- és átmeneti puffert. 43170-43192: élő sejtek számolása össze. 43195-43207: a képernyőn levő populáció elmentése a 45000-es címtől kezdődően. 43210-43224: visszairja az elmentett populációt a képernyőre.

Megjegyzés: a program a képernyő keretét mérgezőnek itéli!

Az 1. lista a BASIC főprogramot tartalmazza, amelyiket ha begépelte, akkor vedd fel magnóra, a SAVE "életjatek" LINE10-zal.

A 2. listán levő programmal lehet bevinni a gépbe a gépi kódú részt. Ha sikeresen betöltötted, akkor vedd fel a BASIC rész után magnóra a SAVE "mcode" CODE 43000, 224-gyel. Ezután betöltve a BASIC programot automatikusan indul és behívja a gépi kódú részt és az életjáték megkezdődik.

A 3. lista a gépi kódú program byte-jai, amit a 2. listán közölt programmal lehet a gépbe bevinni.

Mező Gyula,
1117 Bp. Irinyi J. u. 42.

2. lista

```

10 CLS
20 PRINT AT 3,5:"Gépi kod beolvaso"
30 LET sum=0
40 FOR i=43000 TO 43224
50 PRINT AT 6,5:i
60 INPUT "Ertek:";ertek
70 POKE i,ertek
80 LET sum=sum+ertek
90 NEXT i
100 IF sum<>21282 THEN PRINT AT 10,0
FLASH 1:"Hibas adatok! Ird ujra!"

```

3. lista

43000:	14	20	33	33	160
43005:	6	30	205	39	168
43010:	35	16	250	35	35
43015:	13	32	243	33	182
43020:	162	17	0	160	1
43025:	192	2	237	176	205
43030:	107	13	62	2	205
43035:	1	22	17	0	160
43040:	1	192	2	205	60
43045:	32	201	17	32	0
43050:	55	62	229	237	82
43055:	22	0	205	131	168
43060:	35	205	131	168	43
43065:	43	205	131	168	225
43070:	229	35	205	131	168
43075:	43	43	205	131	168
43080:	225	229	213	17	32
43085:	0	25	209	205	131
43090:	168	35	205	131	168
43095:	43	43	205	131	168
43100:	225	229	126	213	17
43105:	192	2	25	209	254
43110:	32	40	19	62	2
43115:	186	40	9	60	186
43120:	40	5	62	32	119
43125:	225	201	62	42	119
43130:	225	201	62	3	186
43135:	40	246	24	239	126
43140:	254	42	40	1	201
43145:	20	201	0	0	0
43150:	33	0	160	1	130
43155:	5	62	32	119	35
43160:	11	120	177	32	247
43165:	201	0	0	0	0
43170:	1	0	0	33	0
43175:	160	17	36	3	126
43180:	254	42	32	1	3
43185:	35	27	122	179	32
43190:	244	201	0	0	0
43195:	33	0	160	17	200
43200:	175	1	36	3	237
43205:	176	201	0	0	0
43210:	33	200	175	17	0
43215:	160	1	36	3	237
43220:	176	195	21	168	0

PRIMÓRA

A program PRIMO A-32 - A-64-es gépeken futtatható. A legkisebb memóriakapacitású gépen is még kb. 500 byte szabad memóriaterület marad (nagyobb gépeken persze több), így a program esetleg módosítható, bővíthető. A 200-230-as sorok begépelésénél különösen ügyelni kell, nehogy hibás adatot írjunk be, mert ez a program indítása után tönkretelheti egész addigi munkánkat. Ezért célszerű beírás után azt kimenteni szalagra vagy diskre, majd ezután indítani a programot, így az nem vesz el ha hibáztunk. A program a 80.85-ös sorban "megnézi", hogy milyen címtartományban van a képernyő tartalom, a gép stack területe, majd ennek megfelelően állítja be a gépi kódú rész változóit, illetve meghatározza annak betöltési címét. A memória felosztása így a következő lesz: legfelül a képernyő tartalom, alatta esetleg egy bővítő program (pl.: CDOS), majd a string terület, a stack, ez alatt kb. 5,5 Kbyte munkaterület a gépi kódú program számára, végül maga a gépi kódú program. A BASIC program és a gépi kódú rész között pedig szabadon felhasználható memóriaterület található.

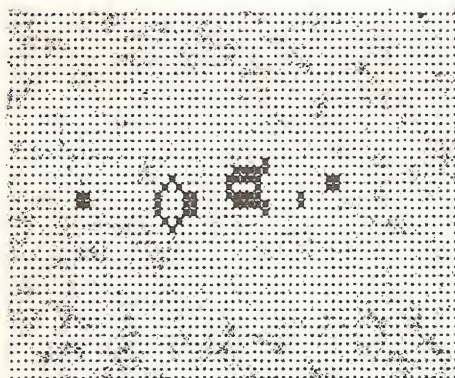
A 90-120-as sorokban bejelentkezik a program és ismerteti a legfontosabb tudnivalókat. Innen a vezérlés a 25-ös sorra adódik.

A 25-75-ös sorokban történik a kiinduló ábra felrajzolása. Ennek megkönnyítésére megjelenik egy raszter, ami a képernyőt 64*45 mezőre osztja. Azt a helyet, ahová éppen rajzolhatunk (vagy éppen törölhetünk onnan), a mező négy sarkának villogása jelzi. Lépní a nyílak segítségével lehet, akár átlósan is, és közben az érintett mezőbe rajzolhatunk az 'R' betű lenyomásával. Törölni hasonló módon a 'T' betűvel lehet. A játékok tényleges indítása a 'RETURN' lenyomásával történik. Ennek hatására a vezérlés a 10-es sorra adódik. A képernyőről törlődik a raszter, csak egy keret marad meg, és a sejtautomata megkezdheti életét. A képernyő tetején megjelenik a generációváltások száma. Egy-egy váltás között kb. 0,5 másodperc telik el. Amennyiben a kereten kívül is rajzolunk sejteteket, akkor azok mint állandó források szerepelnek, mivel a program csak a belső 62*43-as területen számol új állapotot. A program működése során először a puffer területen hozza létre az új állapotnak megfelelő ábrát, majd azt átmásolja a képernyő területre.

A 15-ös sorban történik a klaviatúra figyelése. 'V' hatására visszatér a program a rajzoló üzemmódba, így módosíthatjuk az aktuális ábrát, az 'U' betűre újra indul a játék, míg a 'K'-val kiléphetünk a programból. Bármelyik másik billentyű lenyomásának idejére leáll a generációváltások sorozata.

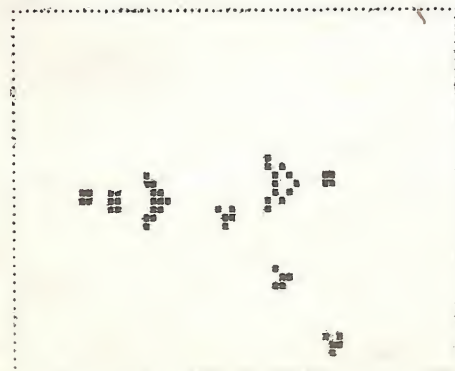
Mindenkinek jó játékot kívánok!

KÁLVIN TAMÁS



Az induló ábra
...és a 78. generáció

78



```

5 GOTO80
10 IFNOTPOINT(4*N+1,4*M+2)THENGOSUB75
15 POKE16452,128:A$=INKEY$:IFA$=""THEN20ELSEIFA$="V"THEN30ELSEIFA$="U"THEN25ELSE
IFA$="K"THEN50ELSE15
20 I=CALL(CI):L=L+1:PRINT#0,0,L:GOTO15
25 CLS:PRINTCHR$(6):L=0
30 N=31:M=22:I=CALL(CI+286)
35 GOSUB70
40 ON(INP(20)AND1)+2*(INP(22)AND1)+3*(INP(55)AND1)GOTO45,50,10:GOTO55
45 FORI=0TO2:FORJ=0TO2:SET(4*N+I,4*M+1+J):NEXTJ:NEXTI:GOTO55
50 FORI=0TO2:FORJ=0TO2:RESET(4*N+I,4*M+1+J):NEXTJ:NEXTI
55 GOSUB75
60 IFPOINT(4*N+1,4*M+2)THENGOSUB70
65 N=N+(INP(61)AND1)-(INP(57)AND1):M=M+(INP(1)AND1)-(INP(15)AND1):N=N-64*(N<0)+6
4*(N>63):M=M-45*(M<0)+45*(M>44):GOTO35
70 SET(4*N,4*M+1):SET(4*N,4*M+3):SET(4*N+2,4*M+1):SET(4*N+2,4*M+3):RETURN
75 RESET(4*N,4*M+1):RESET(4*N,4*M+3):RESET(4*N+2,4*M+1):RESET(4*N+2,4*M+3):RETU
N
80 OUT0,8:POKE16443,8:CLERE50:DEFINTA-Z:IFPEEK(16561)>127THENJ=PEEK(16562)ELSEJ=
PEEK(16562)-1
85 I=PEEK(16458)+1:J=J-2:CI=J-1:CLS:PRINT#4,9,CHR$(2)"ELET JATEK"CHR$(18):PRINT
#7,10,"Inta: KÁLVIN TAMÁS":PRINT#9,17,1986:L=256*(CI+256*(CI>127))+80:GOSUB200:C
I=L
90 CLS:PRINT"A Pont mozgatasa a nyilakkal tortenik. Egysszerre több irányban i
s tortenhet a Pont mozgatasa."PRINT:PRINT"R" - rajzolas az adott pozicioba
T - torles az adott poziciobol
95 PRINT"Rajzolni es torolni is lehet a Pont mozgatasa kozben."PRINT:PRI
NT"RETURN" - a jatek inditasa."PRINT
100 PRINT"V" - visszateres a rajzolasi uzemmodba U - a jatek ujrainditasa
K - kilapes a programbol"
105 PRINT#15,5,CHR$(4)"BARMELYIK GOMBBA LAPOZHATSZ"CHR$(20)
110 IFINKEY$=""THEN118
115 CLS:PRINT"Barhely masik gomb leallitja a jatekot, mindaddig, amig a
gomb le van nyomva."PRINT#12,5,CHR$(4)"BARMELYIK GOMBBA INOITHATSZ!"CHR$(20)
120 IFINKEY$=""THEN120ELSE25
200 POKE1,33,160,J,6,0,62,J+22,112,35,188,194,87,C1,33,0,I+1,17,160,J,1,32,0,126
,230,240,18,9,235,9,235,122,254,J+22,194,102,C1,33,31,I+1,17,191,J,126,230,31,18
,9,235,9,235,122,254,J+22,194,122,C1
205 POKE1+56,221,229,14,43,17,160,J,33,128,1,253,33,0,I+1,217,33,128,I+1,217,6,3
1,175,221,111,203,118,202,166,C1,60,203,86,202,174,C1,60,221,44,44,203,118,202,1
83,C1,60,221,44,217,203,118,202,190,C1,60
210 POKE1+110,203,86,202,198,C1,60,221,44,44,203,118,202,207,C1,60,221,44,217,25
3,203,0,118,202,216,C1,60,253,203,1,118,202,224,C1,60,254,3,202,241,C1,254,2,194
,246,C1,253,203,0,86,202,246,C1
215 POKE1+161,62,14,205,86,C1+1,28,221,125,203,86,202,255,C1,60,217,203,86,202,6
,C1+1,60,217,253,203,0,86,202,15,C1+1,60,253,203,1,86,202,23,C1+1,60,254,3,202,4
0,C1+1,254,2,194,45,C1+1,253,203,1,118,202,45,C1+1
220 POKE1+216,62,224,205,86,C1+1,253,44,5,194,157,C1,197,1,97,0,9,253,9,235,9,23
5,5,217,1,97,0,9,217,193,13,194,155,C1,17,0,I+1,33,160,J,1,96,21,237,176,221,225,2
01
225 POKE1+262,197,213,235,79,182,119,125,198,32,111,121,182,119,125,198,32,111,1
21,182,119,235,209,193,201,33,224,1,17,97,0,62,1+22,60,14,17,6,31,113,44,16,252,
13,113,25,188,194,119,C1+1,33,224,I+22,1,0,32
230 POKE1+316,113,44,16,252,201
235 FORI=0TO2000:NEXTI:RETURN
500 POKE16443,136:OUT0,136:CLS

```

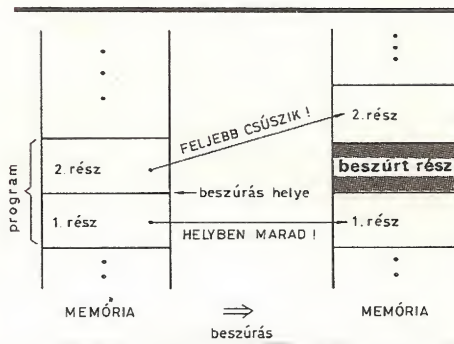


Nemrégiben levelet kaptunk egy olvasónktól (nevét kérte, hogy ne közöljük), aki a következő tapasztalt **Spectrumjén futó 1.8. verziójú BETA BASIC bővítéssel:**

100 KEYIN "20"+CHR\$245+"3"
C Nonsense in BASIC, 100:1
hibajelzéssel állt le. Ha GOTO 100 utasítással újraindította a programot, az hibátlanul továbbfutott. Hogy lehet ez, hogy egyszer jó a sor, máskor ugyanaz nem? És egyáltalán ebben a sorban mi lehet hibás? – kérdezi olvasónk.

A válaszokhoz egy kissé korábbról kell kezdeni a dolgokat. A BETA BASIC a normál BASIC-hez képest (egyebek között) a következő plusz utasításokkal rendelkezik: DELETE, KEYIN. Mindkettő alkalmas a tárban levő program módosítására a futás ideje alatt! Csakhogy ezek használatával vigyázni kell! **A dolgok pontos megértéséhez először részletesen leírjuk működésüket:** DELETE n TO m utasítás törli a programból az n-től m-ig terjedő sorokat (n-et és m-et is beleértve). Ha n-et nem adunk meg, akkor a program elejétől

Tehát a kitörölendő rész úgy „tűnik el”, hogy a mögötte levő részek lejjebb kerülnek a memóriában. Ha maga a DELETE utasítás a második részben van, akkor a művelet elvégzése után az interpreter azon a címen folytatná a programszöveg karaktereinek olvasását, ahol előzőleg abbahagyta. A DELETE utasítás azonban a címet megfelelően korrigálja. Hasonló jellegű, de pontosan fordított probléma jelentkezik a KEYIN parancs használatánál is, ha új programsort iktatunk be.



pillanatban hívjuk meg, mivel a BETA BASIC a szubrutin és a PROCEDURE hívásokkor is azt a címet jegyzi meg ahova vissza kell térni, azaz ezzel nem oldottuk meg a problémát. Ez utóbbi probléma a 3.1. verziójú BETA BASIC-nél is megmaradt, bár ott már a KEYIN utasítás végrehajtása helyes. (Nem áll le ilyen hibajelzéssel.)

Ha már a hibáknál tartunk ...

E sorok írója tapasztalt (szerencsére csak egyetlen programsorral) egy hibát az interpreter editorában, amelyet később sikerült reprodukálni, tehát valóban, az editor tartalmazza azt a csapdát, amit most leírok (a 3.1. verzióban!).

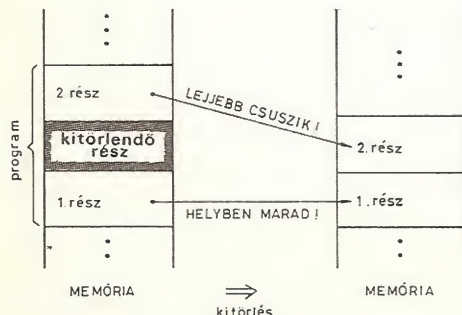
Történt ugyanis, hogy egy programban egy sort módosítani akartam. El is jutottam odáig, hogy lehívtam a sort editálni a képernyő alsó részére. A soron belüli kurzormozgatók során a kurzor egyszer eltűnt! (Ennek részleteiről annyit, hogy az utolsó, soron belüli kurzormozgás lefelé történt.) A továbbiak során a sorban nem sikerült újra előcsatolni a kurzort, de az ENTER gomb megnyomására a rendszer elszállt. A BETA BASIC-et és a programot újra betöltve

A BETA BASIC

kezdve töröl (a 0. sort meghagyja!); ha m hiányzik, a program végéig.
KEYIN a\$

A kulcsszó után írt stringkifejezést úgy tekinti, mintha azt a billentyűzetten írtuk volna be: BASIC utasítássornak tekinti; szintaktikailag ellenőrzi, és beilleszti a programba, ha az sorszámmal kezdődik, ill. azonnal végrehajtja, ha nem.

Tehát összegezve: a két utasítás segítségével lehetőség van programsorok törlésére, átírására, beírására programfutás közben. A magyarázathoz át kell még gondolnunk, hogy ez az egész művelet hogyan is történik. Mivel a BASIC (éppúgy a BETA BASIC is) interpreter típusú nyelv, ez azt jelenti, hogy nem készül lefordított, ún. tárgyprogram, hanem a forrásprogramban (azaz a programszövegben) karakterről karakterre halad az interpreter, és ha számára értelmezhető karaktersorozatot talál, akkor valamilyen hasznos műveletet (Pl. kiírás, beolvasás, értékdadás stb...) végrehajt; ha értelmetlen, akkor hibajelzést ad és újabb parancsot vár. Ebből a fent leírt műveletsorból a „karakterről karakterre haladás” a lényeg most számunkra. Nézzük meg ugyanis, hogy a **DELETE utasításnak mi lesz a hatása:**



Azaz a program 2. része feljebb került, és mivel az interpreter ott folytatja a programszöveg olvasását, (tehát nem korrigálja a felcsúszást) ahol abbahagyta, ezért ott valószínűleg nem egy értelmes utasítást fog találni (lásd még példánkat is).

Összefoglalva tehát: a beszűrő utasítások, ha az éppen végrehajtás előtti részre vonatkoznak, általában hibajelzést okoznak, bár sem szintaktikai hiba nincs sehol, sem végrehajtáskor nem történik semmi rendellenes, csak az interpretert „vertük át”.

A szövegben hivatkoztunk egy példára. Annak igazolására, hogy „direkt” lehet írni olyan programot, mely ilyen esetben sem áll le hibajelzéssel, egyúttal a „feljebb” csúszást bebizonyítottuk, írtunk egy ilyen példaprogramot. Ajánljuk részletes áttanulmányozását és megértését! (Nem állítjuk, hogy sok értelme van, de szerintünk jól magyaráz.)

Példa (melyben a 40. és 50. sor kétszer hajtódik végre)
10 DELETE 20 TO 20
20 REM
30 LET A\$="20"+CHR\$245+CHR\$34+"123456789"+CHR\$34
40 :PRINT"abcde"
50 KEYIN A\$
(A 10-es sor csak azt biztosítja, hogy többszöri futtatás esetén az elején ne legyen 20-as sor.)

A „hibák” elkerülésére sajnos még azt sem tudjuk ajánlani, hogy a program elejére gyűjtjük össze PROCEDURE-be, vagy szubrutinba a KEYIN utasításokat, és azokat a szükséges

ugyanazon az útvonalon mozgatva a kurzort, újra eltűnt.

Ha valaki ugyanezen csapdát el szeretné kerülni, csak azt tudom javasolni, hogy ha editálás során a kurzor eltűnik, akkor a sort hívjuk le a képernyő aljára újra (EDIT gomb, CAPS SHIFT+1), majd más útvonalon haladjunk a kurzorral (ez esetben valószínűleg a lefelé/felfelé mozgások okozzák a gondot, valószínűleg elég az is, ha második alkalommal ezt nem használjuk.) A programunk és BETA BASIC rendszerünk ez esetben üzemképes marad.

Halász Péter

Bármely program

bonyolultsága

addig fokozódik,

amíg túl nem nő

programozója képességein!

(Murphy törvénykönyve)

PROGRAMMA. IÁNLAT:

SPECTRUM FLASH
Commodore 16-on, Plusz/4-en

A program lehetővé teszi, hogy Spectrum villogást használjunk C 16-on. Ez abban különbözik a megszokott villogástól, hogy nem a "látható-nem látható" módon villog, hanem egyszerűen inverz a kírás, egyszer nem. A villogást a ! utasítással idézhetjük elő. Az utasítás szintaktikája:

! x-koordináta, y-koordináta, "szöveg".
Az x-, és y-koordináta lehet konstans és változó, a szöveg csak idézőjelek közé tett karaktersorozat lehet, változó nem. Pl. !12,12, "KISMACSKA".
A villogtatást a megszakító rutin átírásával oldjuk meg. A megszakítórutint úgy írhatjuk át, hogy a megszakítás vektorba beírjuk a mi megszakításunk kezdőcímét (alsóbyte, felsőbyte), majd a mi megszakítás rutinunk végére beírunk egy ugró utasítást az eredeti megszakító rutinra. Ez JMP \$ CE0E.
A megszakítás vektor helye: \$0314. A megszakító rutinunknak az a feladata, hogy az összes 01-es színekű karaktert inverzbe váltsa. Azért a 01-est, mert ez a nyolc fekete közül az egyik, és azért nem a 0-ás színekűt, mert alapállapotban ezzel jelentkezik be a gép.

Új utasítást létrehozhatunk többféle módon. Pl. a CHRGET rutin átírásával. Én egy másik lehetőséget mutatok be. A módszer lényege az, hogy az interpter amikor hibát talál, akkor a \$0300-an található hibavektor által mutatott címre ugrik. Ha mi ezt átírjuk, akkor minden SYNTAX ERROR kírása előtt ellenőrizhetjük, hogy az általunk választott utasítást találta-e a gép. Ezután beolvashatjuk az adatokat, ellenőrizhetjük az esetleges hibákat, és végrehajthatjuk az új utasítást. A hibarutinunkban ellenőriznünk kell, hogy SYNTAX ERROR-e? Ha nem, akkor az eredeti hibáüzenetre ugunk. Ha igen, akkor ellenőrizzük, hogy felkiáltójel-e? Ha nem, akkor az eredeti hibarutinra ugrás, ha igen, akkor a mi rutinunkra ugrás. A hibavektor eredeti értéke: \$8686.

A programot mindig a BASIC memória végére helyezzük, és a BASIC memória felső határát a program alá állítjuk. (A BASIC memória felső határát az 55-56-os címeken állíthatjuk.) Ezért a program megkérdezi, hogy 16 K-s, vagy 64 K-s gépen dolgozunk-e?

A program kiszámítja az ellenőrző összeget, ha ez nem egyezik, akkor hibásan gépeltük be a DATA sorokat.

Ifj. Gulyás László

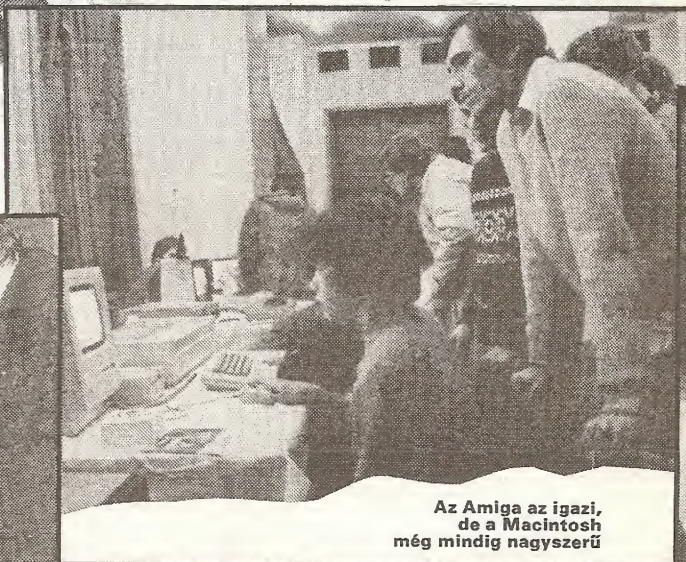
```

1 REM *****
2 REM *
3 REM * SPECTRUM FLASH C-16-ON *
4 REM *
5 REM * KESZITETTE: IFJ. GULYÁS L. *
6 REM *
7 REM * SZENTES, 1986. IX. 5. *
8 REM *
9 REM *****
10 INPUT " 16K / 64K (1/2) ": K: IFK<100K>2 THEN 10
20 IFK=2 THEN T1=126: T1=127: ELSE T1=62: T1=63
30 C=T*256+90: FOR I=0 TO 289: READ A: D=0+A: IFA=62 THEN A=T
36 IFA=63 THEN A=T1
40 POKE C+I, A: NEXT I: IF C<30914 THEN PRINT "HIBA A DATA SOROKBAN!!!!!!": END
50 POKE 55, 58: POKE 56, T: SYST1*256+21: CLR
1000 DATA 224, 11, 240, 3, 76, 215, 62, 173, 19, 63, 208
1010 DATA 251, 32, 165, 4, 201, 32, 208, 6, 32, 223, 62
1020 DATA 76, 102, 62, 201, 33, 208, 94, 32, 129, 157, 224
1030 DATA 40, 176, 92, 138, 72, 32, 145, 148, 32, 230, 62
1040 DATA 32, 129, 157, 224, 25, 176, 77, 104, 158, 24, 32
1050 DATA 240, 255, 32, 145, 148, 32, 223, 62, 32, 165, 4
1060 DATA 201, 34, 240, 50, 201, 0, 240, 46, 201, 13, 240
1070 DATA 42, 201, 32, 144, 29, 201, 130, 144, 12, 201, 149
1080 DATA 144, 21, 201, 156, 144, 4, 201, 160, 144, 13, 166
1090 DATA 202, 164, 205, 32, 210, 255, 32, 237, 62, 76, 204
1100 DATA 62, 32, 210, 255, 32, 223, 62, 76, 153, 62, 76
1110 DATA 214, 139, 162, 11, 76, 134, 134, 162, 14, 76, 134
1120 DATA 134, 230, 59, 208, 2, 230, 60, 96, 199, 59, 208
1130 DATA 2, 198, 60, 96, 142, 16, 63, 169, 8, 141, 17
1140 DATA 63, 192, 0, 240, 20, 24, 169, 40, 109, 16, 63
1150 DATA 141, 16, 63, 169, 0, 109, 17, 63, 141, 17, 63
1160 DATA 136, 208, 236, 169, 1, 141, 241, 9, 96, 0, 0
1170 DATA 169, 0, 141, 19, 63, 141, 20, 63, 169, 90, 141
1180 DATA 0, 3, 169, 62, 141, 1, 3, 169, 50, 141, 20
1190 DATA 3, 169, 63, 141, 21, 3, 96, 238, 20, 63, 173
1200 DATA 20, 63, 201, 32, 208, 60, 160, 4, 162, 0, 189
1210 DATA 0, 8, 201, 1, 208, 8, 189, 0, 12, 73, 120
1220 DATA 157, 0, 12, 232, 208, 238, 238, 66, 63, 238, 73
1230 DATA 63, 238, 78, 63, 136, 208, 224, 169, 8, 141, 66
1240 DATA 63, 169, 12, 141, 73, 63, 141, 78, 63, 169, 0
1250 DATA 141, 20, 63, 173, 19, 63, 73, 128, 141, 19, 63
1260 DATA 76, 14, 206
2000 PRINT "I": !12,12, "EZ A SPECTRUM FLASH!!!!!"

```

Szerkesztői megjegyzés: sajnos a program 64 K-s 16-oson, vagy Plusz/4-esen való futtatásakor csak 26 Kbyte szabad memóriaterület marad. Ez kétség-telenül levon a program értékéből, de úgy gondoltuk mégis érdemes köz-
lésre.

December 13-14-én a Műszaki Egyetemen másodszor rendeztük meg karácsonyi bulinkat. Meg kell mondanunk, hogy a siker minden várakozásunkat felülmúlta. A két nap alatt 5000 ember fordult meg nálunk, az 50 programcsereberére kialakított asztal a vasárnap délelőtt néhány órájától eltekintve állandóan tele volt. A bemutatóteremben is nagy élet volt. Sikerként bérelnünk egy videó kivetítőt, s így az Amiga, a Macintosh és a többi gép képét nagyban láthatták a beszélgetések, bemutatók közben. A tavalyi, városszéli indítás után, tehát úgy érezzük, hogy méltó helyen, méltó sikerrel zártuk az 1986-os évet. Emlékeztetőül azoknak, akik ott voltak, s kedvesnőjének jövőre azoknak, akik megint nem jöttek el, ime néhány pillanattfelvétel.



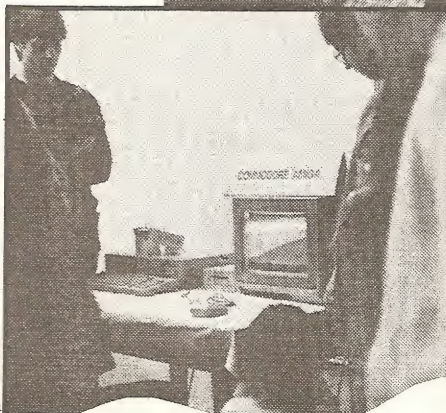
Az Amiga az igazi,
de a Macintosh
még mindig nagyszerű



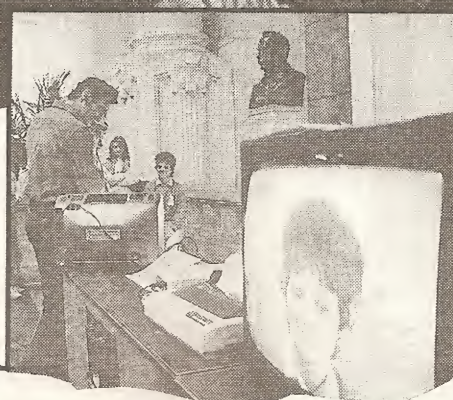
A bemutatóteremből
a szemünk láttára
indult útjára az éterbe
a Spectrum programok
jele.
Aki nekik voltak vételi
tapasztalataik, kitölthet-
ték a vételminőségről
készült felmérő lapot.



- No, ehhez mit szólsz?
Mit tudsz adni érte?



A bemutató terem slágere
természetesen az Amiga volt!



Tessék leülni, mosolyogni,
s először a képernyőn,
azután a nyomtatón
jön elő a mosolya.
A képdigitalizáló rendszerek
még újdonságnak számítottak...

Karácsony!

A csúcstartó több mint ezer lemezzel érkezett az előre kibérelt csereberaszalhoz



Talán jövő ilyenkor már használhatjuk is valamire az új magyar terméket. Lesz mit fölívni vele!



Látkép a magasból. Ezt a „panoráma” fotót képdigitalizálóval készítettük, printelttük képriportunk számítógépesítésére!

Te jó ég!

Impozáns látványt nyújtott az aula. Ez csak egy sor a hárból...

A BEÉPÍTETT RUTINOK HÍVÁSA

Az operációs rendszer beépített rutinjai az egy byte-os RST 30h utasítással hívhatók, melynek kódja 247, azaz 0F7h. A szükséges paramétereket a rutinok a BC és DE regiszterpárokból várják, illetve itt adják vissza az eredményt. Ezenkívül visszatérés-kor az A regiszterben zérus van, ha a rutin hiba nélkül lefutott, egyébként pedig a hibakód, melynek jelentése a TVC Kezelési Útmutató 44. oldalán található. A többi regiszter tartalmát nem változtatja a rutin hívás.

Az operációs rendszer az RST 30h utasítást követő byte-ot tekint a hívott rutin kiválasztási kódjának, úgy, mintha itt nem egy, hanem kettő byte-os utasítás állna, és a második byte lenne az operandus. Gondoskodik arról, hogy visszatérés-kor a kiválasztási kódot követő utasításra kerüljön a vezérlés.

A kiválasztási kód bitejének jelentése:

b7 művelet iránya: 0=output, 1=input

b6-b4 funkcióosztály kódja: 0=video, 1=billeltyűzet, ..., 7=Kernel
Kijelöli a hozzárrendelési táblázat egy elemét a b7-tel együtt. A táblaelem alapján az operációs rendszer megállapítja, hogy melyik logikai eszköznek kell a rutin hívást kiszolgálnia.

b3-b0 konkrét rutin sorszáma: 0-15

A Kernel kivételével minden logikai eszköz első három rutinjának rögzített a funkciója:

0: IT-kiszolgálás

Az operációs rendszer IT-kezelést végző része hívja, felhasználói programból nem ajánlott hívni.

1: karakter input-output

Egy karaktert küld ki vagy olvas be. A karakter helye a processzor C regiszterében.

2: karakter sorozat input-output

Adott memóriaterület byte-jaira sorban elvégzi az 1. számú rutint, feltéve, hogy a memóriacím nem haladja meg a HI MEM rendszerváltozó értékét. A karakterek számát a BC regiszterpárban, a memóriaterület kezdőcímét a DE regiszterpárban várja. HI MEM változó kétbyte-os, címe 2841=0B19h. A rutinokat a logikai eszközök szerint csoportosítva ismertetjük. Megadjuk a használt rendszerváltozókat, a rutinok paraméterezését és a rövid működési leírást. A hívási kód értéke a hozzárrendelési táblázatok eredeti értéke esetén érvényes. Ha a táblázatokban módosítottunk, akkor ahhoz a hívási kód megfelelő három bitjét is hozzá kell igazítani.

FONTOS:

A beépített rutinok hívásakor a 0. lapon a rendszerterületet tartalmazó felhasználói RAM-nak kell lenni, a Stack Pointer pedig a 0. vagy 1. lapra kell, hogy mutasson!

VIDEO RUTINOK

A video rutinok alapvetően a képernyőre rajzolást végzik. Lehetőséget adnak pontok, vonalak rajzolására, képernyőterület kifestésére, valamint karakterek kirajzolására is a képernyő tetszőleges (!) helyén. A rajzolás egy képzeletbeli tollal történik, ezt mozgatja BASIC-ből a PLOT utasítás is. A toll helyét a BASIC-ben megismert koordinátákban kell megadni: X=0...1023, Y=0...959. A képernyő bal alsó sarka a 0;0 pozíció.

A használt rendszerváltozók a következők:

\$L MODE - 1 byte, címe 2891=0B4Bh

Az alsó két bit a pontkiírás típusát adja: 0=felülírás, 1=logikai és (AND), 2=logikai VAGY (OR), 3=logikai KIZÁRÓ VAGY (XOR). BASIC-ből a SET MODE állítja.

\$L STYLE 1 byte, címe 2892=0B4Ch

Az alsó négy bit a vonaltípust adja. BASIC-ből a SET STYLE állítja.

\$INK - 1 byte, címe 2893=0B4Dh

A színfelbontástól függően az alsó egy, kettő vagy négy bit a tintaszínt (palettakódot) adja. BASIC-ből a SET INK állítja.

\$PAPER - 1 byte, címe 2894=0B4Eh

Mint a \$INK, csak a papírszínre. BASIC-ből a SET PAPER állítja.

\$BORDER - 1 byte, címe 2895=0B4Fh

Az aktuális keretszín adja. A BASIC-beli SET BORDER paraméterének kétszeresét tartalmazza.

\$V FLAG - 1 byte, címe 2896=0B50h

Az alsó két bit a video karakter kirajzolást vezérli: az 1 értékű bit letiltja a 0 értékű bit pedig engedélyezi a karakter megfelelő pontjainak kirajzolását; b0=tintaszínű pontok, b1=papírszínű pontok. A pontok kiírása \$L MODE szerint történik.

A rutinok leírása:

VID IRQ - hívási kód: 0

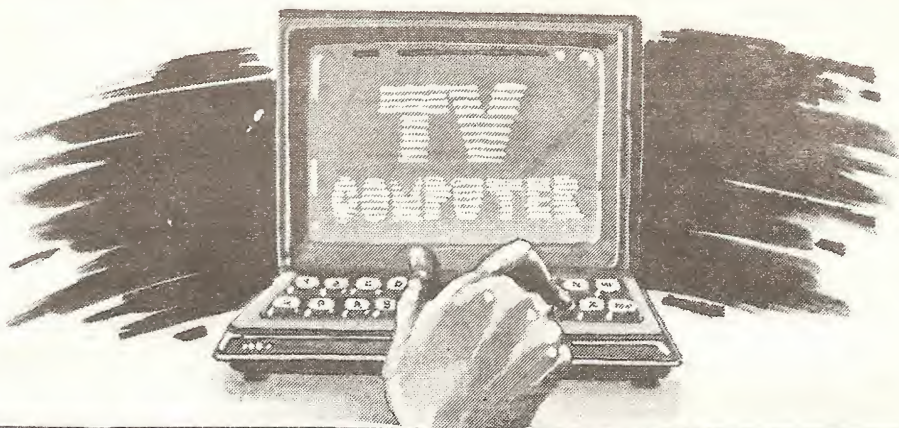
működés: Nincs hatása.

VID CHOUT - hívási kód: 1

input: C=kiírandó karakter

output: A=hibakód

működés:



ELSŐ KÉZBŐL

A TV COMPUTER RÓL

Kirajzol egy karaktert. Az aktuális rajzpozíció lesz a kiírt karakter bal felső sarka. Kiírás után a rajzpozíciót úgy állítja, hogy a következő karakter jobbról kövesse a mostanit. Ha a kép jobb szélén nem fér ki a karakter, akkor tíz pontsorról lejjebb a balszálon folytatja, ha alul nem fér ki, akkor nem ír ki semmit, tehát nem rollozza a képet. A vezérlő karakterek közül csak a CHR\$(10) és CHR\$(13) hatásos, a többire, valamint 223-nál nagyobb kódú karkarakterekre nem reagál.

VID BKOUT - hívási kód: 2

input: BC=kiírandó karakterek száma

DE=első karakter címe

output: A=hibakód

működés: A karakterket egyenként kiírja a VID CHOUT rutinnal.

BTEXT - hívási kód: 3.

input: B=oszlop

C=sor

output: A=hibakód

működés:

A rajzpozíciót beállítja a B;C karakterpozíció bal felső sarkába, ezzel előkészíti a normál szövegsorba való írást a VID CHOUT rutin számára. A képernyő bal felső karakterének koordinátái: 1;1. Ha B vagy C értéke zérus, akkor az a pozíció változatlan marad. Ez a rutin soha nem húz vonalat!

V MODE - hívási kód: 4

input: C=grafikus mód: 0=kétszínű, 1=négyszínű, 2=tizenhatszínű

output: A=hibakód

működés:

Beállítja a grafikus felbontást. Hatása teljesen azonos a BASIC-ben kiadott GRAPHICS utasítással.

CLS - hívási kód: 5

működés: Törli a képernyőt, mint a BASIC-ben kiadott CLS parancs.

BABS - hívási kód: 6

input: BC=X koordináta: 0...1023

DE=Y koordináta: 0...959

output: A=hibakód

működés:

Beállítja a rajzpozíciót az X;Y koordinátákra. Ha a jelképes toll le van téve, akkor vonalat is húz a rendszerváltozóknak megfelelően.

BREL - hívási kód: 7

input: BC=X relatív koordináta

DE=Y relatív koordináta

output: A=hibakód

működés:

Hasonló a BABS rutinhoz, csak itt a megadott koordinátákat hozzáadja az aktuális értékekhez, így kapja az új pozíciót.

BON - hívási kód: 8

működés:

A jelképes tollat leteszi, és kiír egy pontot \$INK és \$L MODE szerint. Ettől kezdve a BABS és BREL rutinok vonalat húznak.

BOFF - hívási kód: 9

működés:

A jelképes tollat felemeli. Ettől kezdve a BABS és BREL rutinok csak pozícionálást végeznek.

FILL - hívási kód: 10 (0Ah)

működés:

Befesti az aktuális rajzpozícióval szomszédos és azonos színű képpontokat a \$INK szerinti színre az "özönvíz" algoritmus szerint. A rutin rekurzív, verem-orientált. Hatása azonos a BASIC-ben kiadott PLOT PAINT utasítással.

DEFC - hívási kód: 11 (0Bh)

input: C=a definiálni kívánt karakter kódja: 128...223

DE=a karaktert definiáló 10 byte kezdőcíme

output: A=hibakód

működés:

Átírja a karakter képét a RAM-ban. Hatása azonos a BASIC-ben kiadott SET CHARACTER utasítással.

PAL - hívási kód: 12 (0Ch)

input: DE=paletta-definiáló 4 byte kezdőcíme

működés:

Definiálja a palettaszíneket a 2 és 4 színű üzemmódhoz. Hatása azonos a BASIC-ben kiadott SET PALETTE utasítással.

Cseh Tibor

A szerkesztő azért van, hogy a lap olyan legyen, amilyenek az olvasói!

KARAKTERKIÍRÁS BASICBŐL VIDEO RUTINNAL

A program bemutatja a videó rutinral történő karakter- és stringkiírást. Ez gyakorlatilag lehetőséget biztosít a képernyőn kis ábrák mozgására, pl. saját tervezésű karakterek (figurák) segítségével.

A program a kezdőszöveg kiírása után egy billentyű leütésére vár (110-199. sor). Az első részben (210-250. sor) a „TV-COMPUTER” stringet kiírja 16 színű módban minden sor elejére, majd az első sorból a karakterek egyenként elvándorolnak az alsó sor jobb oldalára, miközben beletörölnek a képernyőn levő szövegbe (260-340. sor, 1. ábra). Amikor az egész string összeáll, akkor függőlegesen felfelé mozog az első sor (410-440. sor), majd a sor végéről a sor elejére, és egy billentyű leütésére vár (510-590).

A második rész bemutatja, hogy kiírhatók külön a karakter tintaszínű és papírszínű pontjai (610-750. sor, 2. ábra)

Cseh Tibor

```

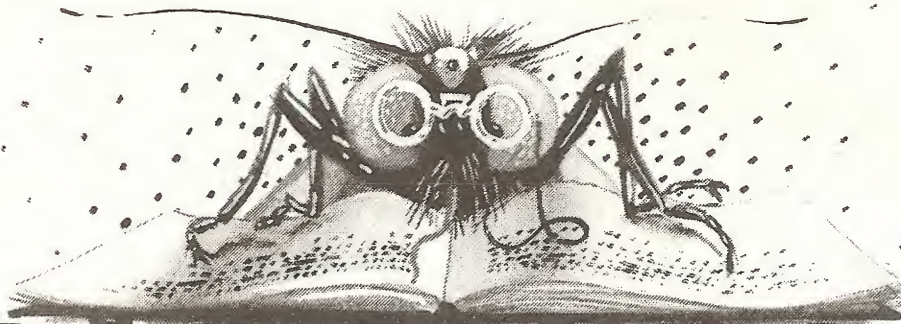
10 !
20 !-----!
30 !
40 ! TV-COMPUTER
50 !
60 ! Karakterkiírás a beépített
70 ! VIDEO rutinral
80 !
90 !-----!
100 !
110 GRAPHICS 4:RANDOMIZE
120 PRINT AT 5,4:"Karakterkiírás a beépített"
130 PRINT AT 8,8:"VIDEO rutinral"
199 GET !<--- vár egy billentyűre
200 !-----!
202 ! egy karakter mozog
204 !-----!
210 GRAPHICS 16
220 T$="TV-COMPUTER":T=LEN(T$)
230 FOR I=1 TO 24
240 : SET INK 9+RND(7):PRINT AT I,1:T$;
250 NEXT I
260 DX=(16-T)*64/230
270 FOR I=T TO 1 STEP -1
280 : SET INK 9+RND(7)
290 : X=(I-1)*64-DX
300 : FOR J=956 TO 36 STEP -4
310 : X=X+DX
320 : PLOT X,J:PRINT #0:T$(I);
330 : NEXT J
340 NEXT I
400 !-----!
402 ! string mozog
404 !-----!
410 FOR J=36 TO 956 STEP 4
420 : SET INK 9+RND(7)
430 : PLOT X,J:PRINT #0:T$;
440 NEXT J
500 !-----!
510 FOR J=X TO 0 STEP -4
520 : SET INK 9+RND(7)
530 : PLOT J,956:PRINT #0:T$;
540 NEXT J
599 GET !<--- vár egy billentyűre
600 !-----!
602 ! V_FLAG bemutatása
604 !-----!
610 V_FLAG=2896
620 GRAPHICS 4
630 TINTA=1:PAPIR=2
640 PRINT:PRINT "Csak a tintaszínű pontok"
650 PRINT "kiírása: POKE 2896,2";:POKE V_FLAG,2
660 FOR I=0 TO 31
670 : SET INK 0:PLOT 32*I,836-4*I
680 : SET INK TINTA:PRINT #0:"T";
690 NEXT I
700 PRINT AT 9,1:"Csak a papírszínű pontok"
710 PRINT "kiírása: POKE 2896,1";:POKE V_FLAG,1
720 FOR I=0 TO 31
730 : SET PAPER PAPER
740 : PLOT 32*I,556-4*I:PRINT #0:"P";
750 NEXT I
800 !-----!
999 GET:PRINT AT 20,1;

```

1. ábra

2. ábra

KERAVILL MEV
MELEKTRONIKAI
MÁRKABOLT 
 BP.V. MŰZEUM krt. 11.
MIKROELEKTRONIKA:
A JÖVŐ A JELENBEN.
 ★★★★★★★★★★★★★★
 FÉLVEZETŐK,
 INTEGRÁLT ÁRAMKÖRÖK,
 MIKROPROCESSZOROK
 ÉS CSATLAKOZÓK.
 SZAKTANÁCSADÁS, CSOMAGKÜLDŐ SZOLGÁLAT.



K Ö N Y V M O L Y

Homonnay Péter: **Angol-magyar számítástechnikai szótár** Novotrade, 142 o., 74 Ft

(A kishoztár azoknak a TPA-11 felhasználóknak nyújt segítséget, akik gépük szakirodalmát eredetiben szeretnék olvasni.)

Az MPS 1000-es pontmátrix nyomtató – Felhasználói kézikönyv Novotrade, 192 o., 180 Ft

(A kötet a nyomtató üzembehelyezésétől, a nyomtatás alapismereteitől kezdve a különböző üzemmódokat mutatja be. A függelék az MPS 1000-es jellemzőit ismerteti.)

Herrmann: **A VC 1541-es lemezegység javítása és karbantartása**

DATA BECKER – Novotrade, 120 o., 300 Ft
(A műszaki kézikönyv a floppy hibaelhárítási lehetőségeit írja le közérthető formában.)

Kampow: **BASIC gyakorlatok a Commodore 64-esen**

(A kezdőknek szóló kötet a C 64 által használt BASIC-V2-be vezeti be az olvasót, jól áttekinthető mintaprogramok segítségével.)

Újabb programfejlesztő rendszerek ESZR gépekre Szerk.: Pásztor János SZÁMALK, 216 o., 102 Ft

(Az „Operációs rendszerek időosztásos üzem-módjai” c. kötet folytatása a QUOTA-II és a GUTS rendszerekről nyújt áttekintést.)

Dr. Ada-Winter Péter-Ada Winter Dávid: **A ZX Spectrum Hardverleírás** – Gépi kódú programozás

Műszaki Könyvkiadó, 360 o., 57 Ft

(A könyv a széles körben elterjedt gép belső felépítésének ismertetése után egy nagy terjedelmű rajzolóprogram megírásán keresztül vezet be a Z80 assembler használatába.)

Dr. Ury László: **Commodore 16, Plus/4, C 64, C 128** Információs kártya – Mikroprocesszorok 65xx, 75xx, 85xx LSI ATSZ, 68 o., 61 Ft

(Programozást segítő kézikönyv a géptípusok jellemzőinek táblázataival.)

Sinclair Spectrum játék és program II. LSI ATSZ, 110 o., 98 Ft.

(Az első kötethez hasonlóan e könyv nagy részét is játékleírások teszik ki, de itt is olvashatók programírást könnyítő szoftverek ismertetései, így a MEGA BASIC V 4.0, illetve hatféle compiler leírása.)

Erdős Iván: **Commodore Plus/4, C-16, C-116 ROM lista**

LSI ATSZ, 284 o., 248 Ft

(A gyakorlati felhasználók számára jól használható ROM listát kiegészíti a keresztreferencia-táblázat, valamint a lista készítésének módját leíró függelék.)

Tóth Viktor: **A Commodore 16-os belső felépítése**

Novotrade, 432 o., 99 Ft

(A processzor-utasításokat, KERNAL-rutinokat bemutató kötet terjedelmes függelék a C 16 ROM listáját tartalmazza.)

BASIC zsebkönyv Szerk.: Kiss Ádám, Kiss Balázs

MÉDEA Kiszövetkezet, 139 o., 99 Ft

(A kis könyv tizenhárom elterjedt géptípus Basic-kulcsszavainak használatát mutatja be, a mellékletként csatolt referencialapok pedig az egyes gépekre vonatkozó ismereteket foglalják össze.)

Informatika Franciaországban – ma Statisztikai Kiadó, 207 o., 150 Ft

(A „társadalom számítógépesítésével” foglalkozó, 1978-ban közzétett Nora-Minc jelentése óta lezajlott fejlődésről, és annak társadalmi következményeiről számol be a kötet.)

Voss: **Bevezetés a statisztikai számítá-sokba C-64-esen**

Novotrade, 225 o., 348 Ft

(Az alapvető statisztikai számítások gépre vitelét segíti a könyv.)

Mi fér az zsebünkbe?

Három apró kis könyv jelent meg a közel-múltban – egy még novemberben, a másik kettő pedig decemberben –, amelyek a számítástechnikával különböző szinten foglalkozók számára zsebkönyv formájában szándékoznak ismereteket közölni.

– **Homonnay Péter: Angol-magyar számítástechnikai szótár**

Homonnay köteté tételezi fel a legtöbb előképzettséget: az előszó szerint azoknak kíván a segítségére lenni, akik a TPA-11 gépcsalád gépein, illetve ESZR gépeken programoznak, és naprakészen szeretnék követni a szakiro-

olvasása közben ne kelljen folyton a nagyszótárt nyálazni.

Ennek a célkitűzésnek eleget tesz a kötetecske – mely a legkisebb terjedelmű a három között. Ennek ellenére a szerkesztéssel kapcsolatban felvethető egy-két kifogás.

Az első – amin a profi számítástechnikusok nyilván csak derűlnek –: sok az olyan címszó, amelynek fordításában, magyarázatában is angol szavak szerepelnek, így a tájékoztal-nabb olvasó kénytelen ezek jelentését is fel-lapozni. Így pl. az ADT (Application Design Tool) értelmezéseként a következő olvasható:

Datatrieve állomány leíró eljárása (alkalma-zási szerkesztő eszköz). Aki azonban nem tudja, mit jelent a „datatrieve”, az kénytelen ezt a szót is megkeresni, hogy megtudja: adatlekérdező rendszerről van szó.

A datatrieve jelentése nyilván sokak számára egyértelmű. Vannak viszont ritkábban használt angol kifejezések is, melyek szintén az értelmező részben szerepelnek – és ezeknél sem utal semmi arra, hogy önálló címszóként is megtalálhatóak a szótárban.

Ezzel szemben sok olyan helyen olvasható a „lásd még:...” megjegyzés, ahol arra sem-mi szükség. Így a „hashmark” fordításaként ezt olvashatjuk: numerikus jel (#) lásd még: number sign. Fellapozva a „number sign” címszót: számjel (#) lásd még: hashmark. A két definíció lényegében ugyanaz, az egy-másra hivatkozás felesleges. Sokkal jobb en-nél pl. a kötőjel (–) meghatározása: dash character: gondolatjel, kötőjel (–) [szinoní-mája: hyphen]; és a hyphen-t már meg sem kell nézni, hiszen nyilván hasonló olvasható ott is.

A kötőződéseknek is beillő kifogások ellenére mindazoknak ajánlható a kötet, akiket idáig hiányos nyelvtudásuk tartott vissza szakmai tájékozottságuk bővítésétől.

– **Mi micsoda magyarul a számítá-s-technikában? Szerk.: Kis Ádám**

A Kis Ádám által szerkesztett könyv – alcíme: Mikroszámítógépes értelmező szótár – a Membrán-sorozatban jelent meg, vagyis népszerűsítő, ismeretterjesztő kötet. Az elő-szó is hangsúlyozza, hogy Akárcinek, Min-denkinek szól, nem szakkönyv, előképzettség nélkül is haszonnal forgatható.

Az első rész az „Angol-magyar számítá-s-technikai kishoztár” – bár itt helytállóbb lett volna a „Számítástechnikai értelmező szó-tár” cím, hiszen a címszavak is magyarul szerepelnek, betűrendben, és csak zárójelben követi őket angol megfelelőjük. Ezután két angol-magyar szójegyzék következik: az első az általános számítástechnikai fogalmak fordítását adja meg, a második pedig az adatát-vitellel kapcsolatos szakkifejezéseket. A füg-gelék a legelterjedtebb BASIC-kulcsszavak értelmezését adja meg.



dalmat. Mivel a magyar nyelvű fordítások ál-talában jelentős késéssel (gyakran elavulva) jelennek meg, a szerző célul tűzi ki, hogy a szakmabeli olvasóval megismertesse azt a néhány száz szakkifejezést, melyek segítségével ez az irodalom megérthető. Eppen ezért sok olyan kifejezést is felvesz a szójegyzékbe, amelyek közhasználatúak, nem tartoznak a szűken vett számítástechnikai nyelvezetbe, de célszerű ismerni őket, hogy a szakcikkek

E kötet is teljesíti feladatát – képes úgy szólni egy általános olvasóközönséghez, mely valamelyest már tájékozott a számítástechnikában, hogy stílusa, nyelvezete eközben nem szájbarágós. Felesleges a szerkesztő mentegőzése az előszóban amiatt, hogy a népes szerzőgárda tagjai egymástól eltérő stílusban írták meg szócikkeiket, mert nem érzékelhető zavaró felfogásbeli különbség. Egyetlen ellenvetés csak a könyv, illetve első fejezetének címével szemben tehető: ezek alapján azt várná az ember, hogy egy, az előzőekben ismertetett Novotrade-kiadványhoz hasonló kétnyelvű szótárt kap kézbe. Itt azonban inkább az értelmezés kap hangsúlyt, nem pedig az idegen nyelvű kifejezések lefordítása.

A kötet jelölésrendszere egyértelmű, jól használható, így a könyv alkalmas arra is, hogy



a számítástechnikával aktívan foglalkozni szándékozó olvasók ne csak egyetlen fogalom meghatározását keressék meg alkalmanként, hanem hogy szócikkről szócikkre kalandozhassanak benne.

–BASIC zsebkönyv Szerk.: Kiss Ádám, Kiss Balázs

A MÉDEA Kiszövetkezet emlékeztetőül szánja kiadványát azoknak, akik már tudnak BASIC nyelven programozni, de időnként utána szeretnék nézni, hogy egy-egy lehetőség hogyan valósítható meg a rendelkezésükre álló géptípuson. A BASIC kulcsszavait betűrendben, egy-egy oldalon ismerteti a könyv, megjelenve azt is, hogy a szerzők által legelterjedtebbeknek tartott tizenhárom géptípus közül melyeken használhatóak. A lapok alján üres rovat „Az én rendszeremben:” címmel, ahová ki-ki beírhatja, hogy milyen eltérést tapasztal az általa használt gép működése és a kötetben leírtak között. A könyvet kiemelhető referencialapok egészítik ki, melyek az elemzett géptípusok kulcsszavait tartalmazzák.

A koncepció dicséretre méltó – a megvalósítás már kevésbé. Itt a kifogások jóval súlyosabbak, mint az előző két kötet esetében. Rengeteg a tárgyi tévedés. Hogy csak néhányat említsünk, melyek a legelterjedtebb gépekkel kapcsolatosak: honnan vették a szerzők (szerkesztők), hogy a Commodore-gepeken, vagy a Spectrumon van ERROR utasítás? hogy a Spectrumon van GET? INSTR? A FIX-nél miért szerepel a Spectrum és a C-64, és az INT-nél miért nem? Ami pedig a legfurcsább: a RUN parancsnál csak

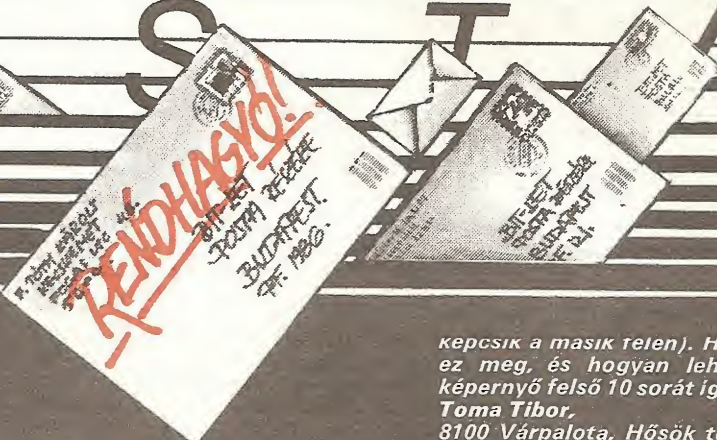
hat gépet sorolnak fel a tizenhárom közül – a többinél vajon hogy indítható a program? A hibákat még hosszasan lehetne sorolni. Érdekes módon a referencialapokon már általában helyesen szerepelnek a kulcsszavak géptípusponként. Maradjunk mindenesetre annyiban, hogy a legtöbb gépkönyv vagy kézikönyv, mely egy-egy géptípus használatát írja le, jóval pontosabb információt nyújt annak BASIC-nyelvről is – rendszerezettebb, áttekinthetőbb formában. Sajnos, éppen a legalacsonyabb ismeretszintet megcélzó zsebkönyvben hemzseg a legtöbb hiba. Sajnos, mert aki épphogy megtanult BASIC-ül, és egy ilyen rosszul használható könyvet kap a kezébe, annak elmegy a kedve attól, hogy megpróbáljon eljutni a számítástechnika mélyebb megismeréséig.

Tallér József

1986-ban megjelent, számítástechnikával foglalkozó könyvek a címek betűrendjében:

Halász: **Alapismeretek a C-64 mikroszámítógép használatához** – Novotrade, 97 o., 140 Ft
Bakó: **Alkalmazói software (C-64)** – Novotrade, 203 o., 105 Ft
Homonnay: **Angol-magyar számítástechnikai szótár** – Novotrade, 142 o., 74 Ft
Lócs: **A BASIC és a kíváncsi** – Feladatgyűjtemény – Tankönyvkiadó, 236 o., 45 Ft
Major-Valovics: **A BASIC feladatok tükrében** – Tankönyvkiadó, 160 o., 43 Ft
Kampow: **BASIC gyakorlatok a Commodore 64-esen** – DATA BECKER – Novotrade, 240 o., 300 Ft
BASIC programozási gyakorlatok a ZX Spectrum-hoz – Ipari Informatikai Központ, 272 o., 250 Ft
BASIC zsebkönyv – Szerk.: Kiss Ádám, Kiss Balázs – MÉDEA Kiszövetkezet, 139 o., 99 Ft
Bevezetés a Basic nyelvbe 1. – Ford.: Kigyós Erzsébet – Novotrade, 132 o., 226 Ft
Voss: **Bevezetés a statisztikai számításokba C-64-esen** – DATA BECKER – Novotrade, 225 o., 348 Ft
Szilassy: **C programozási útmutató** – Szabvány és ajánlás – LSI ATSZ, 58 o., 64 Ft
C-64 adatfeldolgozási lehetőségei – Ipari Informatikai Központ, 136 o., 230 Ft
Angerhausen-Bruckmann-English-Gerits: **A C-64 belső felépítése** – DATA BECKER – Novotrade, 316 o., 355 Ft
C-64 gépi nyelvű programozásának gyakorlata, a 6510-es µP – Ipari Informatikai Központ, 226 o., 290 Ft
Dr. Ferenczy: **C-64 START** – LSI ATSZ, 167 o., 170 Ft
C-128 alkalmazói segédlet – Ipari informatikai Központ, 276 o., 400 Ft
Tóth: **A Commodore 16-os belső felépítése** – Novotrade, 432 o., 99 Ft
Dr. Ury: **Commodore 16, Plus/4, C-64, C-128** – Információs kártya – Mikroprocesszorok 65xx, 75xx, 85xx – LSI ATSZ, 68 o., 61 Ft
Farkas-Bálint: **Commodore 64 file-kezelés és input-output** – LSI ATSZ, 142 o., 119 Ft
Bodor-Gerő: **A Commodore 64 programozásának gyakorlata** – Alapismeretek 1. SZÁMALK, 160 o., 55 Ft
Vadnai: **Commodore 64 programozói zsebkönyv** – Novotrade, 75 o., 139 Ft
Commodore 64 ROM programja – Ipari Informatikai Központ, 210 o., 325 Ft
Commodore 64 Sw II. Ipari Informatikai Központ, 276 o., 270 Ft
Commodore 64 Sw III. – Ipari Informatikai Központ 200 o., 292 Ft
Erdős: **Commodore Plus/4, C-16, C-116 ROM lista** – LSI ATSZ, 284o., 248 Ft
Commodore plus/4 – A beépített programok kezelése – Novotrade, 154 o., 99 Ft
Commodore plus/4 – Felhasználói kézikönyv – Novotrade, 160 o., 99 Ft
Easy file-től a MASTER 64-ig – Adatfeldolgozó programcsomagok Commodore 64-re – Szerk.: MIKROVILÁG GM. LSI ATSZ 254 o., 185 Ft
Easy script felhasználói kézikönyv – Novotrade, 130 o., 220 Ft
Bencsikné: **Feladatgyűjtemény C-16 számítógépre általános iskolásoknak** – Novotrade, 224 o., 163 Ft
Seres-Fenyő-Balogh: **A FORTH programozási nyelv** – Műszaki Könyvkiadó, 291 o., 84 Ft
Pataki-Tallér: **Fűtési rendszerek számítása személyi számítógéppel** – Műszaki Könyvkiadó, 278 o., 158 Ft
English: **Gépi kódú programozás a C-64-esen** – DATA BECKER – Novotrade, 125 o., 241 Ft
English: **Gépi kódú programozás haladóknak C-64, PC-128** – DATA BECKER – Novotrade, 122 o., 319 Ft
Vitray: **Hetedhét Atari 800 XL** – Novotrade, 151 o., 92 Ft
Pál-Révbiró: **Hetedhét C-16** – Novotrade, 139 o., 59 Ft
Pál-Révbiró: **Hetedhét C-64** – Novotrade, 49 o., 66 Ft

Pál-Révbiró: **Hetedhét C Plus-4** – Novotrade, 145 o., 79 Ft
Horváth-Révbiró: **Hetedhét ZX Spectrum** – Novotrade, 213 o., 99 Ft
18085-es µP család – Ipari Informatikai Központ, 140 o., 203 Ft
Erdős: **IBM PC, XT információs kártya** – LSI ATSZ, 97 o., 180 Ft
Sz. Lukács: **Informatika – Szakközépiskolai példatár HT 10802** – Ifjúsági Kiadó, 102 o., 38 Ft
Informatika Franciaországban – ma – Statisztikai Kiadó, 207 o., 150 Ft
Newman: **Interaktív számítógépes grafika** – Műszaki Könyvkiadó, 491 o., 165 Ft
Gerő-Ila-Mihályfi: **Interface 1 – Microdrive** – SZÁMALK, 131 o., 64 Ft
Ránky: **Ipari robotok programozása és alkalmazása** – Ipari Informatikai Központ, 222 o., 430 Ft
Ismerd meg a BASIC nyelvjeleseit – Commodore 64, Commodore VIC 20, SHARP PC-1500 – Szerk.: Kóhgyi János – Műszaki Könyvkiadó, 277 o., 63 Ft
Dacosta: **A kalendárium irásának rejtelméi** – Műszaki Könyvkiadó, 277 o., 63 Ft
Bozsaky: **Kirándulás a számítógépek szigetére** – Tankönyvkiadó, 96 o., 35 Ft
Dr. Tokodi: **A HLASER mikroszámítógépcsalád** – LSI ATSZ, 144 o., 140 Ft
A LOGO programozási nyelv – Műszaki Könyvkiadó, 317 o., 83 Ft
Fekete: **Matematika és számítástechnika 1.** – Műszaki Könyvkiadó, 205 o., 66 Ft
Fekete: **Matematika és számítástechnika 2.** – Műszaki Könyvkiadó, 273 o., 98 Ft
Mi micsoda magyarul a számítástechnikában – Szerk.: Kis Ádám – Tömegkommunikációs Kutatóközpont, 171 o., 75 Ft
Radnai: **Mikroprocesszor berendezések vizsgálata** – Műszaki Könyvkiadó, 259 o., 70 Ft
Mikroszámítógép-alkalmazási esettanulmányok – Szerk.: Dr. Rózsa Lajos – SZÁMALK, 371 o., 156 Ft
Mikroszámítógép kiállítások tapasztalatai – LSI ATSZ, 273 o., 228 Ft
Dahmke: **Mikroszámítógépek operációs rendszerei** – Műszaki Könyvkiadó, 199 o., 79 Ft
Dr. Dobay: **Mikroszámítógépes programkatalógus** – LSI ATSZ, 273 o., 180 Ft
Szlávi-Zsakó: **Műdszeres programozás** – Műszaki Könyvkiadó, 116 o., 50 Ft
Az MPS 1000-es pontmátrix nyomtató – Felhasználói kézikönyv – Novotrade, 192 o., 180 Ft
Operációs rendszerek időosztásos üzemmodjai I-II. – Szerk.: Gerő Zoltán – SZÁMALK, 510 o., 213 Ft
Bakos: **Pascal PC-eknek** – Műszaki Könyvkiadó, 161 o., 53 Ft
PC 10-PC 20 DOS – Novotrade, 348 o., 1400 Ft
PC 10-20-PC GW BASIC – Novotrade, 428 o.
Bárdos-Körtvélyesi: **Programozási alapeladatok gyűjteménye** – SZÁMALK, 211 o., 101 Ft
A PTK 1050-es zsebszámológép alkalmazása az iskolákban – Szerk.: Appel György – Tankönyvkiadó, 175 o., 27 Ft
Ránky: **Robot kezek, szerszámok és robot kiegészítők** – Ipari Informatikai Központ, 162 o., 290 Ft
Rucc: **Rutinról rutinra** – Bepillantás a Sinclair Spectrum gépi kódú világába – LSI ATSZ, 134 o., 149 Ft
Plenge-Szcepanowsky: **Simon's BASIC gyakorlatok** – DATA BECKER – Novotrade, 225 o., 355 Ft
Sinclair QL felhasználói programok – Ipari Informatikai Központ, 346 o., 557 Ft
Sinclair Spectrum játék és program – Szerk.: Székely László – LSI ATSZ, 248 o., 194 Ft
Sinclair Spectrum játék és program II. – LSI ATSZ, 110 o., 98 Ft
Jánoki-Kocsis: **Számítógépes termelésirányítás** – Műszaki Könyvkiadó, 329 o., 90 Ft
Varga: **Személyi számítógépek kezelése, programozása és alkalmazása** – Terra, 326 o., 137 Ft
Dr. Szentcs: **A szoftverminőség mérése** – SZÁMALK, 247 o., 107 Ft
Bencsikné: **Tanári segédkönyv a Commodore 16 számítógépre általános iskolásoknak** – Novotrade, 63 o., 52 Ft
Angerhausen-English-Gerits: **Tippek és trükkök a C-64-eshez** – DATA BECKER – Novotrade, 186 o., 302 Ft
Weltner: **További tippek és trükkök a Commodore 64-eshez** – DATA BECKER – Novotrade, 187 o., 239 Ft
Bóna-Erényi-Vajda: **Többmikroprocesszoros rendszerek** – Műszaki Könyvkiadó, 313 o., 93 Ft
Tudomány és technika Commodore 64 – DATA BECKER – Novotrade, 263 o., 215 Ft
TV-Basic – SZÁMALK, 383 o., 120 Ft
Újabb programfejlesztő rendszerek ESZR-gepeken – Szerk.: Pásztor János – SZÁMALK, 216 o., 102 Ft
Herrmann: **A VC 1541-es lemezegység javítása és karbantartása** – DATA BECKER – Novotrade, 120 o., 300 Ft
English-Szcepanowsky: **A VC 1541-es lemezegység programozása** – DATA BECKER – Novotrade, 280 o., 355 Ft
Dachsel: **Zenekönyv a Commodore 64-eshez** – DATA BECKER – Novotrade, 151 o., 323 Ft
Dr. Ada-Winter Péter-Ada-Winter Dávid: **A ZX Spectrum – Hardverleírás – Gépi kódú programozás** – Műszaki Könyvkiadó, 360 o., 57 Ft
Ligeti-Szervánszky: **A ZX Spectrum programozása** – SZÁMALK, 199 o., 84 Ft
Bosetti: **ZX Spectrum – Tippek és trükkök** – DATA BECKER – Novotrade, 160 o., 199 Ft



Tisztelt Szerkesztőség!

Van egy ZX Spectrum típusú számítógémem. Ehhez szeretnék venni egy olyan nyomtatót, amely A4-es formátumú normál papírra nyomtat. Kérem, írják meg nekem néhány nyomtató típusát (a csatlakozó interface típusával együtt), amelyek igényemnek megfelelnek.

Előre is köszönöm.

Gaál Béla,

8460 Devecser, Hunyadi u. 5.

Csaknem minden nyomtató hozzákapható megfelelő interface-en keresztül a géphez. Azt tudjuk tanácsolni, hogy a vétel helyén érdeklődjön. Olyan sokféle interface és nyomtató létezik, hogy ezt itt nem tudjuk felsorolni. (Arról nem beszélve, hogy Magyarországon a beszerzés nem könnyű, és nem olcsó, ezért gondoljuk, hogy ön is külföldön akar nyomtatót venni.)

Örülök a TVC ismertetésének. (Mert nekem is ez van.) Ha már van kinek feltenni, lenne néhány kérdésem:

1. A készülő soros illesztő mit tesz lehetővé? Kb. mennyi lesz az ára?

Válasz: ez egy szabványos RS232 kimenetet produkál, szabványos csatlakozóval, 12 V-os feszültségzinttel stb. Napokon belül kapható lesz (lehet, hogy mire ez megjelenik, már el is fogyott.) Ára 2000-3000 között.

2. Készül-e ASSEMBLER monitor, akár EPROMban is?

Válasz: igen készülget, EPROMban. De monitorügyben kis meglepetést a BIT-LET is tartogat önnek!

3. Hogyan áll a TVC OS (Operációs rendszer) könyv ügye?

Válasz: rosszul... A szerzőt Cseh Tibort (lapunk szerzője is) ugyanis 8 hónapra katonai szolgálatra hívták be.

4. Múltatás: Tetszik a BIT-LET, főként ezért járatom az Ötletem. (Nagy ő!) Szívesen venném drágábban, gyakrabbi BIT-LET-tel.

Válasz: kösz...

Füle Sándor

6400 Kiskunhalas Sallai u. 18.

Az október 30-i szám „Könyvmoly” rovatában elmarasztaló cikk jelent meg Bosetti: ZX Spectrum Tippek és trükkök c. könyvéről. A cikk szerzője későinek tartja a könyv megjelenését – nem a magyar fordítást, hanem az eredetiét –, mert szerinte a felhasználók már túljutottak azon a szinten, amit a könyv megcélzott.

A tények azonban azt mutatják, hogy a szeptember közepén piacra került könyv az üzletekből egy hónap alatt elfogyott.

A könyv magyar nyelvű kiadását indokoltan érezzük, úgy gondoljuk, hogy a főként családoknál, és nem munkahelyeken levő ZX Spectrum gépeknél fejlődést jelent, ha a tulajdonosok az

eddig főként „lövöldöző” játékkomputer mellett a könyv alapján pl. Mastermind-ot vagy más logikai játékot játszanak.

Nem tartjuk hibának, hogy a könyv szerzője – megemlítve a korlátokat – táblázatok grafikus ábrázolására és adatnyilvántartásra is közül programokat, hiszen az ilyen jellegű feladatok lényegét a kis gépre írt programok is bemutatják. Az ilyen ismeretek megszerzése növeli az általános számítástechnikai kultúrát.

A könyv azért ajánlja a programok beillesztését a fejezetek végén levő listák alapján, mert ezek a ténylegesen futó programok listái, ellentétben a szöveg között – a megértést segítő – részletekben közölt, gépirási és nyomdai hibáknak kitett programrészekkel.

A könyv nagy érdemének tartjuk, hogy a magyar kiadásban minden program működik (ellentétben az eredeti német kiadással), mert ezzel érheti el igazán célját a szerző, a fordító és a kiadó, hogy könnyen juttassa új ismeretekhez olvasóját.

Novotrade Rt. Kiadó Szerkesztősége

Toma Tibor 8. osztályos tanuló vagyok, 3 éve programozok C 64-es gépen. Bár nekünk nincsen, már kölcsöngépen sokat megtanultam, olyannyira, hogy belevágtam egy játékkönyv írásába. SPRITE-okat tudok tervezni, de csak 8-at. Több könyv is kecsegtetett azzal, hogy lehet 16-ot, de hogy hogyan, azt nem írták le. Hogy lehet ezt megcsinálni?

A másik gond a nagyfelbontású képpel van. Tudok ilyet készíteni (multicolor is), ezt egy gépi kódú szubrutin segíti elő, mely Lángos István: A Commodore 64 mikroszámítógép kezelése és programozása című könyvében a 77. oldaltól a 87. oldalig van részletesen leírva. Egy-két kép tervezése után jöttem rá, hogy ezeket jó lenne mozgatni, erről a C 64 régi angol gépkönyvében van is szó, a 166-167. oldalon lévő táblázatban találtam meg ezt a két címet Y SCROLL = 53265, X SCROLL = 53270.

Elkezdtem játszani a számokkal, meg lehet figyelni, hogy 7-ig növekvő sorrendben jobbra mozdul el, majd balra, de azt nem sikerült megoldani, hogy folyamatosan mozgás szemléltessen.

Végül a 3. kérdés: van egy képem, amelynek ponttárolója: 8192-16381-ig van, és szintárolója 1024-2023-ig. Ezt akarom úgy mozgatni, mintha egy hengerre feltekert képet mozgatnánk (mindig visszatér az egyik feléről eltűnő

kepcsik a másik felén). Hogyan oldható ez meg, és hogyan lehet mondjuk a képernyő felső 10 sorát így mozgatni?

Toma Tibor,

8100 Várpalota, Hősök tere 2.

8-nál több sprite elhelyezését csak gépi kódban, a C 64 megszakítás-kezelésének alapos ismerete mellett lehet programozni. Mivel ez komoly munkát igényel, így mi csak az elvét magyarázzuk el, egy lehetséges módot.

A tv képernyőjére a látványt a raszter rajzolja fel, másodpercenként többször is. A C 64-esen van egy raszterszámláló regiszter. Ez éppen a képernyőnek azt a sorát jelöli ki, amelyik éppen rajzolódik. Ha ebbe a regiszterbe egy megfelelő számot írunk, akkor azt a video-chip eltárolja, és amikor a rasztersugár aktuális sora és ez a szám megegyezik, akkor egy programmegszakítást generál, ha az engedélyezve van. A programnak ezt a lehetőséget kell kihasználnia. Amikor a raszter például a képernyő felső felét rajzolja ki, akkor adjuk meg az első 8 sprite pozícióját és alakjának a címét, továbbá beállítjuk a képernyő közepére a raszterszámlálót is, hogy ha odaér, akkor megszakítás generálódjon. Amikor ez megtörtént, újból átvesszük a szerepet és villámgyorsan átkapcsolunk a másik 8 sprite adataira, hogy a képernyő alsó felén azok rajzolódjanak ki. Mivel ennek rendkívül gyorsan kell megtörténnie, ez csak gépi kódú programmal valósítható meg. Itt újból be kell állítani a raszterszámláló összehasonlító értékét a képernyő tetejére, hogy az első 8 sprite adatait a következő megszakítások vissza tudjuk állítani. A képernyő vízszintes irányú pontonkénti mozgatása szintén csak gépi kódú program segítségével oldható meg. Ezt elvben azzal valósíthatjuk meg, hogy a levelemben említett scroll-regiszter értékét folyamatosan változtatjuk, és amikor a 0-ról 7-re vagy 7-ről 0-ra való váltásnak kell következnie, akkor a képernyőn levő összes karaktert egy pozícióval a megfelelő irányban eltolva átmásoljuk, az egyik széléről eltűnnek a karakterek, a másik szélére pedig kirajzoljuk az új karaktereket.

Ezzel a képernyőnek egy nagyon szép mozgatása érhető el, persze csak karakteres grafikával. A bittérképes üzemmódban annyi pontot kellene átmásolni, hogy még a gépi kód is lassúnak bizonyul a finom mozgatáshoz, és így ebben az üzemmódban gyakorlatilag a scroll használhatatlan, tehát a finomfelbontást definiált karakterekkel próbáld elérni.

A képernyő felső sorának mozgatásához már a fent leírt két elv kombinációja szükséges. Csak megszakítások esetén nem a sprite-okat állítgatjuk, hanem a képernyő pozícióit és a megfelelő pillanatokban való másolást kell gépi kódunkban megoldani.

Van egy C 64 személyi számítógépem, egy ITT televízió és aggódó szüleim. Az ő megnyugtatójukra írom ezt a levelet. Azt szeretném kérdezni, hogy a gépem tehet-e valamiféle kárt a tévében, és ha igen, akkor mi az?

Niedermayer Zoltán,
5742 Elek, Tancsics u. 30.

Válaszunk rövid és reméljük, megnyugtató. A számítógépek semmiféle kárt nem tudnak tenni a tévében, ha a kezelési útmutató szerint használják azokat.

Kiegészítés a narancs díjhoz



Kiegészítés a narancs díjhoz

A novemberi számban közöltük Grósz Attila Narancs díjas RE-NEW-ját a C16-os gépre. A szerző ismét jelentkezett, az alábbi kiegészítéssel!

A közölt RE-NEW a PLUS/4-esen, illetve a bővítővel ellátott 16-oson nem működik minden esetben. A problémát az okozza, hogy ha grafikát használunk, az interpreter a BASIC programot nem \$1000-tól használja, hanem áthelyezi a \$4000-en kezdődő területre.

Ahelyett, hogy folytatnánk Grósz Attila eszmefuttatását, máris kitágtjuk a kört. Minden olyan esetben gond lehet, amikor áthelyeztük a BASIC elejét. De a RE-NEW elve ettől még igaz, az általános megoldás pedig a következő:

RESET után, ha X értékének az éppen aktuális BASIC eleje címet tekintjük:
POKE 43,(X-32768) AND 255
POKE 44,X/256
POKE X,1:RENUMBER

Ha érted,

akkor már elavult!

Bitton posztulátuma

a naprakészen korszerű

elektronikáról.



A GORDIUSZI CSOMÓ...

C 16, C+4 tulajdonosok!

100 éves gondja a Commodore-osoknak, hogy ha JUNOSZTY tv-t akarnak használni, nem tudnak hangot produkálni. Most egy élelmes olvasónk úgy tűnik megoldotta ezt a problémát, s ha nem is tökéletesen, de elnyiszálta a gordiuszi csomót.

JUNOSZTY és más SECAM-os tv-eket nem kell a számítógéphez áthangolni! A számítógép tv-kimeneténél található fémdobozban van egy tolókapcsoló, ennek átkapcsolásával rendes hang is jön a tv-n. A kapcsoló elérhető egy kis csavarhúzóval is a fémdoboz tetejéről. Két állása van G és I jelzéssel. I-be kell kapcsolni pl. JUNOSZTY esetén. Más tv esetén bármikor visszakapcsolható. A „műtét” szakember nélkül is végrehajtható veszély nélkül. A legtöbb gépnél még a garancia se láthatja kárát, mert a gépdoboz csavarjai nincsenek leplombálva.

Kotroczó Béla, 5711 Gyula, Széchenyi u. 43.

A vizsgálatot elvégeztük. Nos a dolog nagyjából működik. Azaz a JVC tévének SECAM módban meg sem nyílik. Egy Elektronika C-401-es ugyan megszólalt, de nagyon zajos volt, a JUNOSZTY 402V azonban egész jól szólt. HIFI-ről persze szó sem esett, de nem is ez volt a cél. Kis vizsgálódásunk eredményeképpen azt is tudjuk, hogy mindennek mi lehet az oka. Az ugyanis, hogy a JUNOSZTY pontatlanul van hangolva! A JVC pontos SECAM-ja kiröhögött bennünket, a Junosztvé azonban kellően átlóg abba a PAL tartományba, amelybe a kis kapcsoló I jelű állásakor belépünk!

Egy szó mint száz, a megoldás javallott!

PROGRAM CSERE-BERE

Commodore 16 programokat cserélek! Rumpf Barnabás, 9400 Sopron, Balfi út 7. **Commodore 64 játék és más hasznos programokat** (másolókat, nyelvi programokat) cserélek magnószalagon. Főleg magnóval is használható programok érdekelnek. Programnévlistát a következő címre: Pozsgai Emil, 8500 Pápa, Bajcsy-Zs. u. 20.

Commodore 64-re cserélek játékokat, rendszerprogramokat, de csak magnóra. Programlistát a következő címre kérem: Lampert Gábor, 8500 Pápa, Beke József u. 14.

Felhasználói és játékprogramokat cserélnék Spectrumra. Pallai József, 6900 Makó, Barcsai u. 10. C16-os számítógépre játékprogramokat cserélnék. Az általam kínált programok saját programok, BASIC-ben íródtak, de élvezhetők. Jakab Zsolt, 1147 Budapest, Czobor u. 124.

Keresem Tv Computer vagy 2+81 tulajdonosokkal a kapcsolatot Program-, és tapasztalatcsere céljából. Oláh Gyula, Debrecen, Ispotály u. 15. 3/15.

Commodore 16 ill. Plusz 4-es anyagokat (programok, felhasználói programok, dokumentáció, „örökletek” stb.) cserélnék. Márkatársak! Irjatok! Benedek Csaba, 2800 Tata-bánya, II. Vadász u. 70. 1/3.

C 64-es, C 16-os és Plusz 4-es programokat cserélek C 64-re. Ebből az állományom 1100 program. Ez utóbbit tudok adni lemezen. Elsősorban felhasználói érdekel. Molnár Tibor, 5530 Vésztő, Eötvös u. 4/A.

ATARI 800XL márkatársakkal keresem a kapcsolatot programcsere és tapasztalatcsere céljából. Pál János, 8484 Nagyalásny, Kossuth L. u. 27.

MSX rendszerű számítógépek levelét várom programcsere céljából. Jancsurák István, 3528 Miskolc, Dráva u. 7.

C 64-es programokat cserélek. FLOPPY-val rendelkezőkkel. Zábó Károly, 2541 Lábatlan, Rákóczi út 284.

ZX Spectrum programokat cserélek! Szabó Attila, 8960 Lenti, Sugár u. 65.

Commodore 16 játékprogramok cseréje. Kerekes András, 1171 Budapest, Oszkár u. 24.

ZX Spectrum és ZX 81-es programok, leírások cseréje miatt keresek ilyen géppel rendelkezőket. Programlistát kérek. Fazekas Attila, 3200 Gyöngyös, Török I. út 26/2.

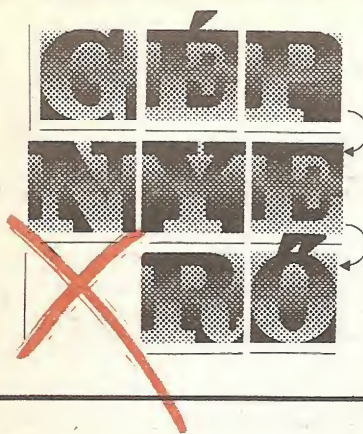
ZX Spectrumhoz felhasználói, és játékprogramokat cserélek. Gedő Tamás, 1165 Budapest, Veres Péter út 121.

ZX Spectrum programokhoz cserepartnert keresek. Tel.: 776-172 Bánffy Szabolcs, 1191 Budapest, Dobó K. u. 19. IX. em. 27.

Commodore 16 programokhoz cserepartnert keresek. Tel.: 575-794. Nyerusay Tibor, 1191 Budapest, Rákóczi u. 1-17. F. 23.

Aquarius home computer tulajdonosokat keresek program cserebere és a gép jobb megismerése céljából. Magyar Ferenc, 2030 Érd, Kankalin u. 20/b.

VC 20 tulajdonos vagyok, s a gépemhez van egy 32 Kbyte-os bővítőm is. **Keresek géptársakat programcsere céljából.** Maró Csaba, 5100 Jászberény, Szövetkezet u. 4/20.



A május-júniusi Gépnyerő értékelése

A feladatok elég könnyűnek bizonyultak, az elsőre (játék Fehér Félix és Fekete Ferenc között) 135, a másodikra (a BNV-s 3. feladat indoklása a megoldások ismeretében) 65 megoldás érkezett, összesen 64 pályázónk küldött be mindkét feladatra megoldást, ebből 34 volt többé-kevésbé helyes (22-25 pont). A kiírásnak megfelelően az első 20 között sorsolunk, azonban mivel 1 db 25 pontos és 19 db 24 pontos megoldás volt, ezen belül részletesebb rangsort nem tudunk felállítani, így a 25 pontos pályázó neve 2-szer, a többieké egyszer kerül majd a „kalapba”. Mivel azonban most a fődíjon kívül egyéb díjakat is sorsolunk, ezért teszünk egy

olyan kikötést, hogy ha a „kétcédulás” pályázónk megnyeri a fődíjat, másik céduláját kiesszük a kalapból.

A nyeremények:

1. díj: Plusz 4 számítógép
 2. díj: 1 doboz floppy disc
 3. díj: 1000 forintos ÁPISZ vásárlási utalvány
 - 4-10. díj: 500-500 forintos ÁPISZ vásárlási utalvány
- A Potenciális nyertesek:**
Földvári Csongor, Budapest (2 cédulával)
Börzsei Katalin, Budapest
Csikós László, Kiskunfélegyháza
Fehér Györgyi, Tiszafüred
Gombos János, Gyula
Horváth Péter, Budapest
Kaczur István, Győr
Kovács Gábor, Vác

Kovács Zoltán, Veszprém
Kozma Benedek, Budapest
Kramarics Géza, Zalaegerszeg
Kruzslicz Ferenc, Tótkomlósi
Kurusa Árpád, Szeged
Peták Tamás, Szolnok
Rónai András, Győr
Szabó Zoltán Tibor, Budapest
Szilvási Margit, Budapest
Tikász Csaba, Budapest
Tunkó Attila, Budapest
Tóth Gábor, Szombathely

A sorsolásra 1987. február 14-én szombat délelőtt 10 órakor kerül sor a Bp. VII. kerületi Almássy téri Szabadidő Központ Compánia névre hallgató számítógépes műhelyében (IV. em. 420.). A sorsolásra minden érdeklődőt, és különösen az érintetteket szeretettel várjuk.

A Harmadgépnyerő 3. feladatának (BIT-LET és Commodoria) megoldása

A feladatra igen sokféle megoldás elképzelhető, mi egy lehetséges variációt közlünk. Előjáróban megjegyezzük, hogy nyilván olyan adatstruktúrát kellett megadni, mely alapján a térképlényege (azaz a járatok) rekonstruálható, s melybe az adatbetöltés nem igényel algoritmuslépéseket, ezt a térkép alapján egy elsőosztályos diák is el kell, hogy tudja végezni.

Az adatokat tároljuk egy $N \times N$ -es táblázatban, melynek sorai (1-N) a városoknak felelnek meg (az első sor a fővárosnak), az oszlopokból a 0-ba fogjuk beírni azt, hogy az illető várost kik lakják (1-bit, 2-letek), a többiek (1-(N-1)) azoknak a városoknak a sorszámaát soroljuk fel, melynek az illető (azaz a sornak megfelelő) várossal járattal vannak összekötve. Ha N-1-nél kevesebb ilyen város van, az utolsó város utáni oszlopba 0 kerül. (A táblázatot – a 0. oszlop kivételével – valóban könnyű kitölteni, csak előtte a térképen be kell sorszámozni a városokat.) Szükségünk lesz az algoritmushoz egy N hosszúságú vektorra is, melybe a már felismert városokat gyűjtjük.

Az algoritmus a következő:

0. lépés: Az első sor 0. oszlopában 1-et írunk (a fővárost bitek lakják), a vektor első helyére 1-et írunk (a fővárost felismertük – bitek lakják), s megjegyezzük, hogy a vektornak 1 eleme van, s hogy ed-

dig a vektor 0. eleménél tartunk (ennek értelmét ld. később).

1. lépés: Vegyük a vektor következő elemét. Ha nincs ilyen, tehát már az utolsónál tartottunk, akkor menjünk az 5. lépésre. Ha van, akkor ennek minden szomszédjára (a táblázatban a sorában lévő város. sorszámkra) hajtsuk végre a 2. lépést. Menjünk vissza az 1. lépés elejére.
2. lépés: Nézzük meg, hogy az illető szomszédot már régebben felismertük-e. Ha igen, hajtsuk végre a 3. különben a 4. lépést.
3. lépés: Nézzük meg, hogy a most soron lévő a vektorból vett város és a táblázatbeli soros szomszédja azonos címkékű-e. Ha igen, menjünk a 6. lépésre, ellenkező esetben ciklus vége, folytassuk az első lépés ciklusát.
4. lépés: Adjunk a szomszédoknak a soron lévő, a vektorból vett város címkéjével ellentétes címkét (ha annak C volt, ennek legyen 3-C), a szomszéd sorszámaát írjuk a vektor végére, jegyezzük meg, hogy annak egygyel több eleme van. Ciklus vége, folytassuk az első lépés ciklusát.
5. lépés: Ha a vektornak N eleme van, akkor készen vagyunk, minden városról sikerült eldönteni, hogy bit vagy let város, az eredményt a táblázat 0. oszlopába tartalmazza. Menjünk a 7. lépésre. Ha kevesebb eleme van a vektornak, az azt jelenti, hogy a többi (a vektornak nem szereplő) városba a fővárosból nem lehet eljutni, így azok milyenségét az adatokból nem tud-

juk megállapítani. Az összes felismerhető város milyensége a 0. oszlopban található. Menjünk a 7. lépésre.

6. lépés: Ellentmondást találunk, a szomszédnak egyszerre kellene bit és let városnak lennie, tehát az adatok hibásak.

7. lépés: Vége.

Annak bizonyítását, hogy az algoritmus jól működik, egyszerűsége miatt az olvasóra bizzuk.

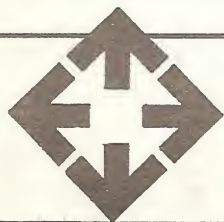
Memória-igény: $N^2 + N + 2$ rekeszre volt szükségünk a mátrix, a vektor és a két pointer tárolására. Ezenkívül az első lépés ciklusához szükség van még 1 ciklusváltozóra, így az összes memória-igény: $N^2 + N + 3$.

Lépcsőszám: ha az i. csúcs szomszédainak számát $d(i)$ -vel jelöljük, akkor nyilván (jó adatok esetén) $d(1) + d(2) + \dots + d(N)$ -szer kell a 2. lépést végrehajtani, ahol alkalmanként 2 vagy 3 műveletet kell elvégezni. Ha a ciklusok adminisztrálásánál minden körben 2-2 lépésnek vesszük, akkor az összlépcsőszám: $4(d(1) + d(2) + \dots + d(N)) + 3N + 4$.

(A 3N-ből 2N a külső ciklus igénye, s a 2. lépésben N-szer mehetünk összesen arra az ágra, ahol 3 lépést kell csinálni, ugyanis minden csúcs csak egyszer, „ismerődik fel”).

A +4 a 0. lépés műveletigénye. Könnyű látni, hogy $d(1) + d(2) + \dots + d(N) = 2M$, így a lépcsőszám: $8M + 3N + 4 \approx N^2$ (vagy N esetén a 3N sokkal kisebb, mint az N^2)! Reméljük a megoldások között találunk ötletesebb, vagy gyorsabb algoritmusokat is. Ha gyöngyszemre lelünk, ígérjük, hogy közreadjuk.

harmad- gép nyerő



PLUS/4 NYERŐ

PLUS/4 NYERŐ 3. FELADATA

99 férfi asztaliteniszező körmerközést játszik.

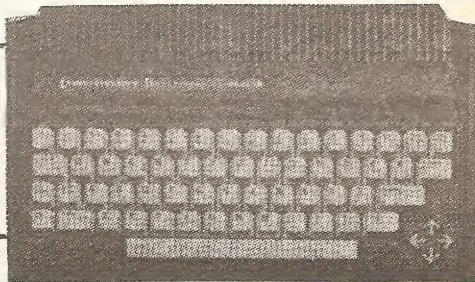
A 3 hónapig tartó viadal végére mindenki mindenkivel pontosan egyszer mérkőzött meg, s persze döntetlen eredmény nem lehetséges. A Versenybizottság feladata lenne ezek után az egyes mérkőzések eredményének (csak az számít, hogy ki győzött, az nem, hogy mennyivel) ismeretében rangsorolni a versenyzőket. A Bizottság már 3 napja ülészik, de képtelen igazságosnak mondható rangsort felállítani. A fő nehézséget az okozza, hogy mind a 99 versenyzőnek 49 győzelme és 49 veresége van.

1. feladat: lehetséges egyáltalán ez? Ha igen, miért nem? (Azaz akár igenlő, akár nemleges válasz esetén részletes indoklást kérünk!)

Ekkor toppan be a bizottsági ülésre Egyenes Elek, aki a problémát meghallva kertelés nélkül, tehát egyenesen kijelenti: igazságos rangsort én sem tudok felállítani, az viszont biztos, hogy fel tudok állítani egy olyan rangsort, ahol mindenki legyőzte az utána következőt. Erre az egyik bizottsági tag megjegyzi, hogy szerinte ebben Elek nem lehet biztos, hiszen nem is ismeri az eredményeket. Elek azonban könnyedén válaszolja: mindegy, hogy a mérkőzések milyen eredményekkel zárultak, sőt még az sem lényeges, hogy mindenkinek 49 győzelme van.

Bármilyen eredménytáblázat esetén lehet a 100 versenyzőt 1-től 100-ig úgy rangsorolni, hogy mindenki legyőzte legalább a rangsorban közvetlenül utána állót. Persze így általában teljesen rossz és igazságtalan rangsor születik, jelen esetben viszont esetleg ez is alkalmazható. A Bizottság erőteljes hitetlenkedésére Elek magyarázatba kezd, s végül sikerül meggyőznie a Bizottság tagjait.

2. feladat: Írják le, vajon hogyan indokolta meg Elek az állítását, hogy még a Bizottságban levő hírhedt Hitetlenkedő Tamást is sikerült meggyőznie igazáról.



Kérjük levégni és a levélre felragasztani! Beküldési határidő: 1987. február 23.